YAMAHA NETWORK BOARD

EtherNet/IP

User's Manual

RCX240

EUS6122100
E122

Ver. 1.00

CONTENTS

Introduction

Introduction i

Safety Precautions (Always read before starting use) ii

Warranty iv

Chapter 1 Outline

1. Features 1-1

2. Mechanism 1-2

3. Installing into the robot controller 1-3

4. Part names 1-4

5. I/O assignments of EtherNet/IP compatible module 1-5

6. EtherNet/IP system connection status transition and robot controller status 1-6

Chapter 2 Connection

1. Confirming the EtherNet/IP compatible module settings 2-1

2. Setting to the EtherNet/IP system specification controller 2-2

2.1 Saving the robot controller data 2-2
2.2 Installing into the robot controller 2-2
2.3 Response when starting up the robot controller 2-2

3. Setting the EtherNet/IP compatible module 2-3

3.1 Making the EtherNet/IP module valid 2-3
3.2 Setting the "Remote_cmd SI05" function 2-4
3.3 Setting the "Output MSG SOW1" function 2-5
3.4 Setting the IP address 2-6
3.5 Setting the subnet mask 2-7
3.6 Setting the gateway 2-8

4. Noise measures 2-9

4.1 LAN cable 2-9
4.2 Mounting the ferrite core 2-9

5. Connecting to the EtherNet/IP system 2-10

5.1 Connecting the LAN cable 2-10

Chapter 3 Communication

1. State when the robot controller power is turned ON 3-1

2. Communication with the master module 3-2

2.1 Receiving data 3-2
2.2 Transmitting data 3-4

3. Direct connection by emulated serialization on parallel DIO 3-5

3.1 Emulated serialization setting on parallel DIO 3-5

4. Referring to the communication data 3-8

4.1 Referring to the data from the programming box 3-8

Chapter 4 Troubleshooting

1. Check items before starting up the EtherNet/IP system 4-1

2. Meanings of LEDs on EtherNet/IP compatible module 4-2

3. Troubleshooting 4-3

3.1 Robot controller front panel LED confirmation 4-3
3.2 RPB (programming box) error display confirmation 4-3
3.3 EtherNet/IP compatible module LED confirmation 4-4
3.4 Confirmation from master module 4-4

4. Error messages relating to EtherNet/IP 4-5

Chapter 5 Specifications

1. Profile 5-1

2. Details of input/output signals 5-3

3. Dedicated input/output signal timing chart 5-6

3.1 Servo ON and emergency stop 5-6
3.2 AUTO mode changeover, program reset and program execution 5-7
3.3 Stopping with program interlock 5-8

4. EtherNet/IP compatible module specifications 5-9

5. EtherNet/IP specifications 5-10

Chapter 6 Appendix

1. Definitions of terms 6-1

2. EDS files 6-4

Remote command guide

1. Remote command format A-1

1.1 Remote command specifications A-1

1.2 Remote status A-2

2. Sending and receiving remote commands A-3

3. Remote command & remote status tables A-4

4. Remote command description A-8

4.1 Special commands A-8

4.1.1 Status reset command A-8

4.1.2 Main robot current position reference command A-8

4.1.3 Sub robot current position reference command A-9

4.2 Category 1 remote commands A-11

4.2.1 MOVE command A-12

4.2.1.1 PTP designation A-12

4.2.1.2 Arch designation A-15

4.2.1.3 Linear interpolation A-18

4.2.1.4 Circular interpolation A-21

4.2.1.5 Direct PTP designation (millimeter units) A-24

4.2.1.6 Direct PTP designation (pulse units) A-27

4.2.2 MOVEI command A-30

4.2.2.1 PTP designation A-30

4.2.2.2 Direct PTP designation (millimeter units) A-33

4.2.2.3 Direct PTP designation (pulse units) A-36

4.2.3 DRIVE command A-39

4.2.3.1 Point designation A-39

4.2.3.2 Direct designation (millimeter units) A-42

4.2.3.3 Direct designation (pulse units) A-45

4.2.4 DRIVEI command A-48

4.2.4.1 Point designation A-48

4.2.4.2 Direct designation (millimeter units) A-51

4.2.4.3 Direct designation (pulse units) A-54

4.2.5 Pallet movement command A-57

4.2.5.1 PTP designation A-57

4.2.5.2 Arch designation A-60
4.2.6 Jog movement command A-63
4.2.7 Inching movement command A-66
4.2.8 Point teaching command A-69
4.2.9 Absolute reset movement command A-70
4.2.10 Absolute reset command A-71
4.2.11 Return-to-origin command A-72
4.2.12 Servo command A-74
4.2.13 Manual speed change command A-75
4.2.14 Auto speed change command A-76
4.2.15 Program speed change command A-77
4.2.16 Shift designation change command A-78
4.2.17 Hand designation change command A-79
4.2.18 Arm designation change command A-80
4.2.19 Point display unit designation command A-81
4.3 Category 2 remote commands A-83
4.3.1 Point-related command A-84
4.3.1.1 Point data definition A-84
4.3.1.2 Point data reference A-86
4.3.2 Point comment-related command A-88
4.3.2.1 Point comment data definition A-88
4.3.2.2 Point comment data reference A-90
4.3.3 Pallet-related command A-92
4.3.3.1 Pallet data definition A-92
4.3.3.2 Pallet data reference A-93
4.3.4 Shift-related command A-95
4.3.4.1 Shift data definition A-95
4.3.4.2 Shift data reference A-97
4.3.5 Hand-related command A-99
4.3.5.1 Hand data definition A-99
4.3.5.2 Hand data reference A-101
4.4 Category 3 remote commands A-103
4.4.1 Static variable-related command A-104
4.4.1.1 Assigning a numerical value to a static variable A-104
4.4.1.2 Assigning a variable to a static variable A-106
4.4.1.3 Arithmetic operation using numerical data on static variable A-108
4.4.1.4 Arithmetic operation using variable on static variable A-110
4.4.1.5 Static variable value reference A-112
4.4.2 Parameter-related command A-114
4.4.2.1 Assigning a value to a parameter A-114
4.4.2.2 Parameter value reference A-116
4.4.3 Point-related command A-118

4.4.3.1 Assigning a point to a parameter A-118
4.4.3.2 Point addition/subtraction A-120
4.4.3.3 Assigning a pallet point A-122
4.4.4 Element assignment command A-124
4.4.4.1 Assigning to a point element A-124
4.4.4.2 Assigning to a shift element A-126
4.5 Category 4 remote commands A-128
4.5.1 I/O port commands A-129
4.5.1.1 Assigning a numerical value to an I/O port A-129
4.5.1.2 I/O port reference A-131
4.6 Category 5 remote commands A-133
4.6.1 Execution program designation A-134
4.6.2 Program execution A-135
4.6.3 Program reset A-136
4.6.4 Program task switching A-137
4.6.5 Program execution information reference A-138
4.7 Category 6 remote commands A-140
4.7.1 Version information reference A-141
4.7.2 Controller configuration reference A-142
4.7.3 Servo status reference A-144
4.7.4 Absolute reset status reference A-145
4.7.5 Current position reference A-147
4.7.5.1 Pulse designation A-147
4.7.5.2 Millimeter designation A-148
4.7.6 Task status reference A-150
4.7.7 Task execution line reference A-151
4.7.8 Message reference A-152
4.7.9 Speed status reference A-153
4.7.10 Arm designation status reference A-154
4.7.11 Arm status reference A-155
4.7.12 Service mode status reference A-156
4.7.13 Point unit status reference A-157
4.7.14 Return-to-origin status reference A-158

I/O command guide

1. I/O command format B-1

2. Sending and receiving I/O commands B-2

3. I/O command list B-3

4. I/O command description B-4

4.1 MOVE command B-4
4.1.1 PTP designation B-4
4.1.2 Linear interpolation B-4
4.2 MOVEI command B-5
4.2.1 PTP designation B-5
4.3 Pallet movement command B-5
4.3.1 PTP designation B-5
4.4 Jog movement command B-6
4.5 Inching movement command B-7
4.6 Point teaching command B-7
4.7 Absolute reset movement command B-8
4.8 Absolute reset command B-8
4.9 Return-to-origin command B-9
4.10 Servo command B-9
4.11 Manual speed change command B-10
4.12 Auto speed change command B-10
4.13 Program speed change command B-11
4.14 Shift designation change command B-11
4.15 Hand designation change command B-11
4.16 Arm designation change command B-12
4.17 Point display unit designation command B-12

Introduction

Contents

Introduction i

Safety Precautions (Always read before starting use) ii

Warranty iv

Introduction

Thank you for purchasing the EtherNet/IP compatible module. This EtherNet/IP compatible module is an option module that allows the YAMAHA robot controller RCX240 to be connected as an EtherNet/IP system slave module. The robot controller explained in this manual refers to the RCX240.

This manual consists of EtherNet/IP compatible module guide (explanation on wiring or communication), remote command guide, and I/O command guide.
For information on other devices such as connecting the master module and sequence programming, refer to the manual for the respective product. For details on operating the robot controller and on the robot program, thoroughly read the controller user's manual and programming manual supplied with the YAMAHA robot controller.

Safety Precautions (Always read before starting use)

Before using this product, be sure to read this manual carefully as well as robot controller user's manual and programming manual. Take sufficient precautions to ensure safety and handle the product correctly.
The cautions given in this manual are related to this product. Refer to the robot controller user's manual for details on the cautions to be taken with the robot controller system using this product.

* The safety precautions are ranked as "WARNING" and "CAUTION" in this manual.

WARNING

FAILURE TO FOLLOW WARNING INSTRUCTIONS COULD RESULT IN SERIOUS INJURY OR DEATH TO THE OPERATOR OR PERSON SERVICING THE PRODUCT.

CAUTION

Failure to follow CAUTION instructions may result in injury to the operator or person servicing product, or damage to the product or peripheral equipment.

NOTE

Explains the key point in the operation in a simple and clear manner.

Note that some items described as "CAUTION" may lead to serious results depending on the situation. In any case, important information that must be observed is explained.

Store this manual where it can be easily referred to, and make sure that it is delivered to the end user.

The EtherNet/IP is a protocol that is jointly controlled by ODVA (Open DeviceNet Vendor Association) and CI (ControlNet International).

■ Precautions for design

WARNING

- **REFER TO THE ETHERNET/IP SYSTEM MASTER MODULE USER'S MANUAL AND THIS MANUAL FOR DETAILS ON THE STATE OF THE ETHERNET/IP SYSTEM AND ROBOT CONTROLLER WHEN A COMMUNICATION ERROR OCCURS WITH THE ETHERNET/IP SYSTEM, ETC. CONFIGURE AN INTERLOCK CIRCUIT IN THE SEQUENCE PROGRAM SO THAT THE SYSTEM, INCLUDING THE ROBOT CONTROLLER WILL WORK SAFELY USING THE COMMUNICATION STATUS INFORMATION.**
- **THE SAFETY CONNECTOR OF THE ROBOT CONTROLLER HAS AN EMERGENCY STOP TERMINAL TO TRIGGER EMERGENCY STOP. USING THIS TERMINAL, PREPARE A PHYSICAL INTERLOCK CIRCUIT SO THAT THE SYSTEM INCLUDING THE ROBOT CONTROLLER WILL WORK SAFETY.**

CAUTION

- **The control line and communication cable must not be bound with or placed near the main circuit or power line. Separate these by at least 100mm. Failure to observe this could lead to malfunctions caused by noise.**
- **The dedicated input of STD.DIO connector provided on the RCX240 controllers will be disabled except for an interlock signal (DI 11). When the external 24V monitor control setting of system parameters is set invalid, the interlock signal (DI 11) will also be disabled.**

■ Precautions for installation

WARNING

- **ALWAYS CRIMP, PRESS-FIT OR SOLDER THE CONNECTOR WIRE CONNECTIONS WITH THE MAKER-DESIGNATED TOOL, AND SECURELY CONNECT THE CONNECTOR TO THE MODULE.**
- **ALWAYS SHUT OFF ALL PHASES OF THE POWER SUPPLY EXTERNALLY BEFORE STARTING INSTALLATION OR WIRING WORK.**
  FAILURE TO SHUT OFF ALL PHASES COULD LEAD TO ELECTRIC SHOCKS OR PRODUCT DAMAGE.

CAUTION

- Use the robot controller within the environment specifications given in the manual. Use in an environment outside the environment specification range could lead to electric shocks, fires, malfunctioning, product damage or deterioration.
- Install the EtherNet/IP compatible module into the robot controller, and securely fix with screws.
- Never directly touch the conductive sections or electric parts inside the controller.
- Accurately connect each connection cable connector to the mounting section.
  Failure to observe this could lead to malfunctions caused by a connection fault.

■ Precautions for wiring

WARNING

- ALWAYS SHUT OFF ALL PHASES OF THE POWER SUPPLY EXTERNALLY BEFORE STARTING INSTALLATION OR WIRING WORK. FAILURE TO SHUT OFF ALL PHASES COULD LEAD TO ELECTRIC SHOCKS OR PRODUCT DAMAGE.
- ALWAYS INSTALL THE TERMINAL COVERS ENCLOSED WITH THE PRODUCT BEFORE TURNING ON THE POWER OR OPERATING THE PRODUCT AFTER INSTALLATION OR WIRING WORK. FAILURE TO INSTALL THE TERMINAL COVER COULD LEAD TO MALFUNCTIONS.

CAUTION

- Tighten the terminal screws within the specified torque range. A loose terminal screw could lead to short-circuiting or malfunctioning. If the terminal screw is too tight, short-circuiting or malfunctioning could occur due to screw damage.
- Make sure that foreign matter, such as cutting chips or wire scraps, do not enter the robot controller.
- The communication cables connected to the EtherNet/IP compatible module must be placed in a conduit or fixed with a clamp. If the cable is not placed in a conduit or fixed with a clamp, the module or cable could be damaged by the cable shifting, movement or unintentional pulling leading to malfunctioning caused by an improper cable connection.
- Do not disconnect the communication cable connected to the EtherNet/IP compatible module by pulling on the cable section.

■ Precautions for starting and maintenance

WARNING

- DO NOT TOUCH THE TERMINALS WHILE THE POWER IS ON. FAILURE TO OBSERVE THIS COULD LEAD TO MALFUNCTIONING.
- ALWAYS SHUT OFF ALL PHASES OF THE POWER SUPPLY EXTERNALLY BEFORE CLEANING OR TIGHTENING THE TERMINAL SCREWS. FAILURE TO SHUT OFF ALL PHASES COULD LEAD TO ELECTRIC SHOCKS, PRODUCT DAMAGE OR MALFUNCTIONING. A LOOSE SCREW COULD LEAD TO DROPPING, SHORT-CIRCUITING OR MALFUNCTIONING. IF THE SCREW IS TOO TIGHT, SHORT-CIRCUITING OR MALFUNCTIONING COULD OCCUR DUE TO SCREW DAMAGE.
- NEVER DISASSEMBLE OR MODIFY ANY OF THE ROBOT CONTROLLER MODULES.
  FAILURE TO OBSERVE THIS COULD LEAD TO TROUBLE, MALFUNCTIONING, INJURIES OR FIRES.
- ALWAYS SHUT OFF ALL PHASES OF THE POWER SUPPLY EXTERNALLY BEFORE INSTALLING OR REMOVING THE ETHERNET/IP COMPATIBLE MODULE.
  FAILURE TO SHUT OFF ALL PHASES COULD LEAD TO ROBOT CONTROLLER TROUBLE OR MALFUNCTIONING.
- WHEN USING THE ROBOT CONTROLLER WITH THE ETHERNET/IP COMPATIBLE MODULE MOUNTED, ALWAYS MOUNT THE ENCLOSED FERRITE CORE FOR NOISE MEASURES ON THE POWER CABLE AS CLOSE TO THE ROBOT CONTROLLER AS POSSIBLE. FAILURE TO MOUNT THIS FERRITE CORE COULD LEAD TO MALFUNCTIONING CAUSED BY NOISE.

CAUTION

The EtherNet/IP system may not function properly if the master module and robot controller power are turned ON simultaneously. Always turn the robot controller power ON after turning ON the power for the master module ON.

■ Precautions for disposal

CAUTION

Dispose of this product as industrial waste.

This manual does not guarantee the implementation of industrial rights or other rights, and does not authorize the implementation rights. YAMAHA shall not be held liable for any problems regarding industrial rights that occur through the use of the contents given in this manual.

2012 YAMAHA MOTOR CO., LTD.

Warranty

For information on the product warranty, please contact your local agent where you purchased your product.

Chapter 1 Outline

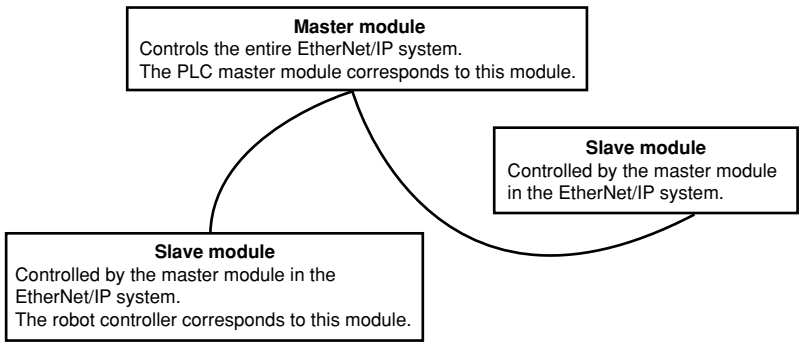
Contents

1. Features 1-1
2. Mechanism 1-2
3. Installing into the robot controller 1-3
4. Part names 1-4
5. I/O assignments of EtherNet/IP compatible module 1-5
6. EtherNet/IP system connection status transition and robot controller status 1-6

1. Features

This EtherNet/IP is an industrial network that is achieved by combining the standard protocol TCP/IP with the higher level protocol CIP (Common Industrial Protocol).
Additionally, since the EtherNet/IP uses the standard protocol TCP/IP and Ethernet as lower level protocols, it can utilize the Ethernet technologies that are widely available in the world.
The EtherNet/IP system connects the robot controllers or distributed input/output systems with dedicated cables to control these units from the master module.
The EtherNet/IP system allows wiring to be reduced.

EtherNet/IP system


Master module
Controls the entire EtherNet/IP system.
The PLC master module corresponds to this module.
Slave module
Controlled by the master module in the EtherNet/IP system.
Slave module
Controlled by the master module in the EtherNet/IP system.
The robot controller corresponds to this module.

23101-S6-00

For details about other units, such as the network settings on the master module side, refer to the user's manual for relevant unit.
Additionally, for details about operation of the controller main unit and robot programming, refer to the user's manuals for controller and programming.

The EtherNet/IP is a protocol that is jointly controlled by ODVA (Open DeviceNet Vendor Association) and CI (ControlNet International).

NOTE

The dedicated inputs of the STD.DIO connector provided on the RCX240 controller will be disabled except for the interlock signal (DI 11). When the "Watch on STD.DO DC24V" of the system parameters is set invalid, the interlock signal (DI 11) will also be disabled.

■ Emulated serialization on parallel DIO

By making the robot controller's internal settings without using a robot program, various I/O devices, such as the sensors and relays mounted on the robot controller's parallel I/O can be controlled from the PLC as if they were EtherNet/IP system I/O devices.

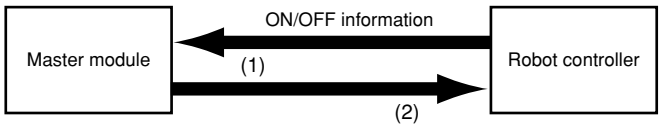
CAUTION

**An emergency stop terminal for hardwire is provided in the SAFETY connector on the robot controller.
For the RCX240, when the EtherNet/IP system is used while STD. DIO is not used (external DC 24V power is not supplied), the "Watch on STD.DO DC24V" of the system parameters must be set invalid. If it is left valid, the STD. DIO interlock signal is enabled, causing an error in the robot operation commands.**

2. Mechanism

This section describes the mechanism of the communication to provide an understanding of how the robot controller and master module operate via the EtherNet/IP system.

Mechanism of communication



23102-S6-00

(1) The robot controller's ON/OFF information is sent to the master module via the network.

(2) The master module's ON/OFF information is sent to the robot controller via the network.

* The robot controller monitors the ON/OFF information at a 10ms cycle.
* The ON/OFF information consists of two words each of dedicated I/O words, 14 words each of general-purpose I/O words as word information, and 16 points each of dedicated I/O points, 96 points each of general-purpose I/O points as bit information.

If the following is executed with the robot program in the robot controller, the bit information will be sent to the master module via the EtherNet/IP system by (1).

SO(20)=1

Conversely, if the following is executed with the robot program, the bit information received from the master module via the EtherNet/IP system will be monitored by (2), and the robot controller will wait for the ON information.

WAIT SI(20)=1

If the following is executed with the robot program in the robot controller, the word information will be sent to the master module via the EtherNet/IP system by (1).

SOW(2)=256

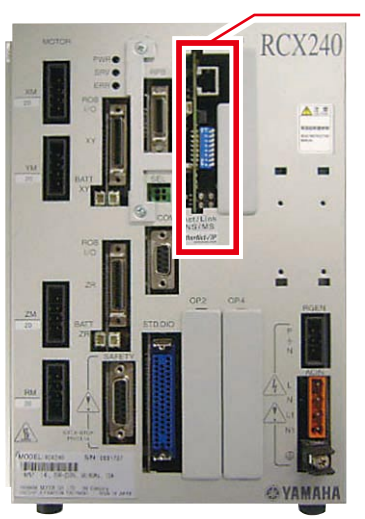
Conversely, if the following is executed with the robot program, the word information received from the master module via the EtherNet/IP system will be substituted in integer variable A% by (2).

A%=SIW(3)

3. Installing into the robot controller

The EtherNet/IP compatible module is installed into an option slot of the robot controller. (Refer to the figure below.)

Installing into the robot controller



23103-S6-00

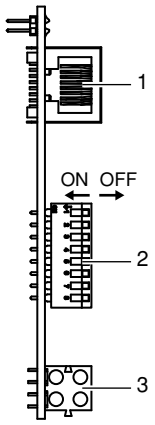
When installing the EtherNet/IP compatible module into the robot controller, strictly observe the following cautions.

* When three or less option boards are installed, do not stack the EtherNet/IP compatible module on other option board. Since the EtherNet/IP compatible module is taller than other option boards, it interferes with other option board.
* When four option boards are installed, install the EtherNet/IP compatible module into the OP.3 slot. Additionally, a custom-ordered front panel is needed. (For details, contact YAMAHA's Sales Department.)
* The CPU BOARD ASSY is applicable only when the ROM capacity is 4MB or more (KX0-M4210-2XX).

4. Part names

This section describes the part names of the EtherNet/IP compatible module to be installed in the robot controller. This module is installed in the option slot of the robot controller.

Part names



23104-S6-00

1. RJ45 connector

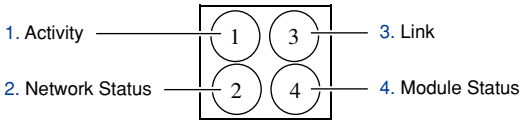
Connect commercially available LAN cable supporting 10Base-T or 100Base-TX.

2. DIP SWITCH

Unused. All switches must be OFF.

3. Status Indicators

Status indicators



23105-S6-00

1. Activity : Flashing green during packet communication.

2. Network Status :

Status	Description
OFF	Power is OFF or no IP address is found.
Lit in green	Detects the online and connects other unit.
Flashing green	Detects the online, but does not connect other unit.
Lit in red	Detects serious error, such as IP address duplication.
Flashing red	Time-out occurs during connection with other unit.
Flashing green/red	Performing the self-test (only when the power is turned ON).

3. Link : Lit in green when the link is detected.

4. Module Status :

Status	Description
OFF	Power is OFF.
Lit in green	Connection with the master is correct.
Flashing green	Connection with the master is not established.
Lit in red	Recoverable minor error is detected.
Flashing red	Unrecoverable error is detected.
Flashing green/red	Performing the self-test (only when the power is turned ON).

5. I/O assignments of EtherNet/IP compatible module

The following describes the correspondence between the serial input/output of the robot controller and the input/output data on the EtherNet/IP. The number of bytes to be assigned to the EtherNet/IP compatible module is 48 byes for input and 48 bytes for output.

Serial output (Robot controller → Master module)				Serial input (Master module → Robot controller)			
Robot controller		Master module		Robot controller		Master module	
Port number		Address		Port number		Address	
	SOW(0)*1	m			SIW(0)*1	n	
	SOW(1)*1	m+2			SIW(1)*1	n+2	
SOD(2)	SOW(2)	m+4		SID(2)	SIW(2)	n+4	
	SOW(3)	m+6			SIW(3)	n+6	
SOD(4)	SOW(4)	m+8		SID(4)	SIW(4)	n+8	
	SOW(5)	m+10			SIW(5)	n+10	
SOD(6)	SOW(6)	m+12		SID(6)	SIW(6)	n+12	
	SOW(7)	m+14			SIW(7)	n+14	
SOD(8)	SOW(8)	m+16		SID(8)	SIW(8)	n+16	
	SOW(9)	m+18			SIW(9)	n+18	
SOD(10)	SOW(10)	m+20		SID(10)	SIW(10)	n+20	
	SOW(11)	m+22			SIW(11)	n+22	
SOD(12)	SOW(12)	m+24		SID(12)	SIW(12)	n+24	
	SOW(13)	m+26			SIW(13)	n+26	
SOD(14)	SOW(14)	m+28		SID(14)	SIW(14)	n+28	
	SOW(15)	m+30			SIW(15)	n+30	
SO0(7~0) *2		m+32	7~0	SI0(7~0) *2		n+32	7~0
SO1(7~0) *2		m+33	7~0	SI1(7~0) *2		n+33	7~0
SO2(7~0)		m+34	7~0	SI2(7~0)		n+34	7~0
SO3(7~0)		m+35	7~0	SI3(7~0)		n+35	7~0
SO4(7~0)		m+36	7~0	SI4(7~0)		n+36	7~0
SO5(7~0)		m+37	7~0	SI5(7~0)		n+37	7~0
SO6(7~0)		m+38	7~0	SI6(7~0)		n+38	7~0
SO7(7~0)		m+39	7~0	SI7(7~0)		n+39	7~0
SO10(7~0)		m+40	7~0	SI10(7~0)		n+40	7~0
SO11(7~0)		m+41	7~0	SI11(7~0)		n+41	7~0
SO12(7~0)		m+42	7~0	SI12(7~0)		n+42	7~0
SO13(7~0)		m+43	7~0	SI13(7~0)		n+43	7~0
SO14(7~0)		m+44	7~0	SI14(7~0)		n+44	7~0
SO15(7~0)		m+45	7~0	SI15(7~0)		n+45	7~0
Reserved. *3		m+46	7~0	Reserved. *3		n+46	7~0
Reserved. *3		m+47	7~0	Reserved. *3		n+47	7~0

m : Start address of the input area assigned to the master module
n : Start address of the output area assigned to the master module

*1: Since this port is used as dedicated command, it cannot be used as general-purpose input/output data.
*2: Since this port is used as dedicated input/output, it cannot be used as general-purpose input/output data.
*3: Reserved area.

NOTE

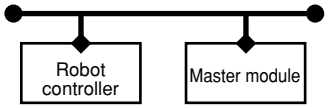
- Each address is 8-bit data.
- SOn() and SIn() are handled as unsigned 8-bit integer data.
- SOW(n) and SIW(n) are handled as unsigned 16-bit integer data.
- SOD(n) and SID(n) are handled as signed 32-bit integer data.
- The upper word and lower word of SOD(n) correspond to SOW(n+1) and SOW(n), respectively.
- The upper word and lower word of SID(n) correspond to SIW(n+1) and SIW(n), respectively.
- In the RCX240, the dedicated inputs of STD.DIO provided on the controller will be disabled except for the interlock signal (DI11). When the external 24V monitor control of the system parameters is set invalid, the interlock signal (DI11) will also be disabled.

6. EtherNet/IP system connection status transition and robot controller status

The EtherNet/IP system specification robot controller always starts the operation in the servo OFF state after the power has been turned ON.

1. Normal state of EtherNet/IP system connection when the robot controller power is turned ON

System connection normal state



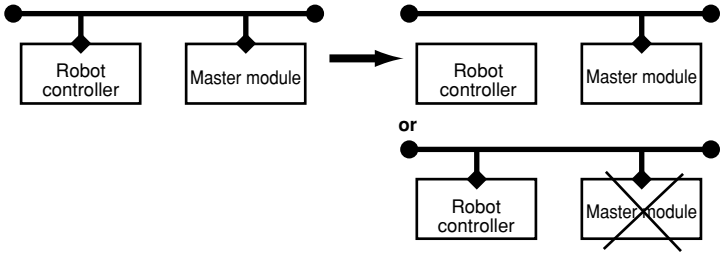
23106-S6-00

- The emergency stop/interlock signals in the EtherNet/IP system are valid.
- When the SAFE mode is enabled, the service mode input signal is made valid with SI (02) in the EtherNet/IP system.
- The emergency stop terminal in the SAFETY connector is valid.
- The interlock signal in the STD. DIO connector is valid unless the external 24V monitor control of the system parameters is set invalid.
- When the SAFE mode is enabled and the external 24V monitor control of the system parameters is left valid, the service mode input signal is made valid with DI(02) in the SAFETY connector.

* The signals in the EtherNet/IP system are sent and received.

2. Transition from the EtherNet/IP system normal connection state to the EtherNet/IP system connection error state

System connection error state (1)



23107-S6-00

- The emergency stop input turns off with SI (00) in the robot controller.
- The interlock signal turns off with SI (10) in the robot controller.
- The service mode input turns off with SI (02) in the robot controller.
- The emergency stop terminal in the SAFETY connector is valid.
- The interlock signal in the STD. DIO connector is valid when the external 24V monitor control of the system parameters is left valid.
- When the SAFE mode is enabled and the external 24V monitor control of the system parameters is left valid, the service mode input signal is made valid with DI (02) in the SAFETY connector.

* The signals in the EtherNet/IP system are not sent or received.

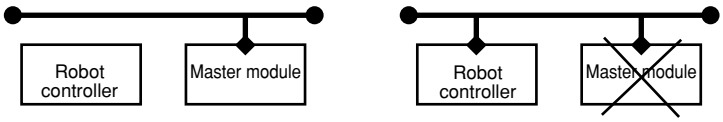
* If the connection to the EtherNet/IP system transits from the normal state to the error state, the EtherNet/IP system connection must be returned to the normal state.

* The signals in the EtherNet/IP system can be sent and received when the EtherNet/IP system connection is recovered to the normal state.

3. EtherNet/IP system connection error state due to following factors when the robot controller power is turned ON.

- It is impossible to connect to the EtherNet/IP system.
- The master module is faulty.

System connection error state (2)



23108-S6-00

- The emergency stop input turns off with SI(00) in the robot controller.
- The interlock signal turns off with SI (10) in the robot controller.
- The service mode input turns off with SI (02) in the robot controller.
- At this time, the emergency stop signal and interlock signal in the EtherNet/IP system turn off. So, the robot controller cannot be operated individually.
  To operate the robot controller individually, change the "Board condition" parameter of the EtherNet/IP module to "INVALID".
- The emergency stop signal terminal in the SAFETY connector is valid.
- The interlock signal in the STD. DIO connector is valid when the external 24V monitor control of the system parameters is left valid.
- When the safe mode is set and the external 24V monitor control of the system parameters is not set invalid, the service mode input signal through the SAFETY connector DI (02) is valid.

* The signals in the EtherNet/IP system are not sent or received.

* If the connection to the EtherNet/IP system transits from the normal state to the error state, the EtherNet/IP system connection must be returned to the normal state.

* The signals in the EtherNet/IP system can be sent and received when the EtherNet/IP system connection is recovered to the normal state.

Chapter 2 Connection

Contents

1. Confirming the EtherNet/IP compatible module settings 2-1
2. Setting to the EtherNet/IP system specification controller 2-2
2.1 Saving the robot controller data 2-2
2.2 Installing into the robot controller 2-2
2.3 Response when starting up the robot controller 2-2
3. Setting the EtherNet/IP compatible module 2-3
3.1 Making the EtherNet/IP module valid 2-3
3.2 Setting the "Remote_cmd SI05" function 2-4
3.3 Setting the "Output MSG SOW1" function 2-5
3.4 Setting the IP address 2-6
3.5 Setting the subnet mask 2-7
3.6 Setting the gateway 2-8
4. Noise measures 2-9
4.1 LAN cable 2-9
4.2 Mounting the ferrite core 2-9
5. Connecting to the EtherNet/IP system 2-10
5.1 Connecting the LAN cable 2-10

1. Confirming the EtherNet/IP compatible module settings

Whether or not an EtherNet/IP compatible module is installed in the robot controller can be confirmed from the RPB programming box (hereafter referred to as "RPB"). The following describes the confirmation steps.

■ Operation

Step 1 ***Select "SYSTEM".***

Press the MODE (ROBOT) key on the RPB to change the function menu, and then press the F4 (SYSTEM) key. The "SYSTEM" screen will appear.

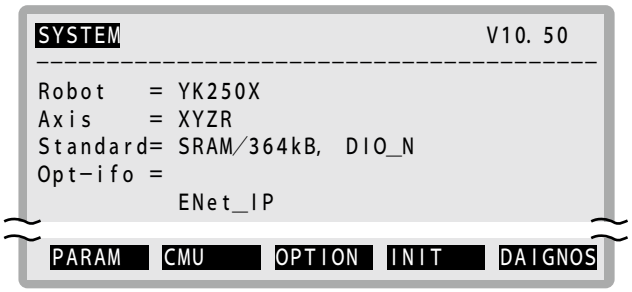
24201-S6-00

Step 2 ***Confirm that the EtherNet/IP module is installed.***

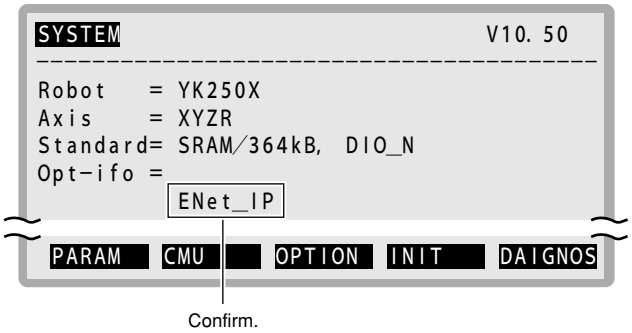
"ENet_IP" is shown in "Opt-ifo" when the EtherNet/IP module is installed properly.

24202-S6-00

▶ **Step 1** "SYSTEM" screen



▶ **Step 2** Confirming that the EtherNet/IP module is installed



2. Setting to the EtherNet/IP system specification controller

When connecting the EtherNet/IP compatible module to an existing robot controller, the EtherNet/IP compatible module must be installed. Check the EtherNet/IP system specifications using the procedure described in "1. Confirming the EtherNet/IP compatible module settings" of Chapter 2 in this guide.

2.1 Saving the robot controller data

Before installing the EtherNet/IP compatible module into the robot controller, be sure to save the data stored in the robot controller into an external memory using the VIP+ software, etc.

2.2 Installing into the robot controller

Install the EtherNet/IP compatible module into the root controller. (Refer to "3. Installing into the robot controller" in Chapter 1 of this guide.) After the EtherNet/IP compatible module has been installed, set the IP address, subnet mask, and gateway for the EtherNet/IP compatible module while referring to "3. Setting the EtherNet/IP compatible module" in Chapter 2 of this guide.

2.3 Response when starting up the robot controller

The robot controller will always start up in the option board setting error state after the EtherNet/IP compatible module has been installed into the robot controller. Follow the steps below to make the settings.

■ Procedure

Step 1 ***Check the controller connections, and then turn ON the power.***

Connect all input connectors on the front panel of the controller, and then turn ON the power.

Step 2 ***Select "YES".***

The message, "change OptionSlot OK?", will appear on the RPB. Select "YES".
After you have selected "YES", the "MANUAL" screen will appear.

24203-S6-00

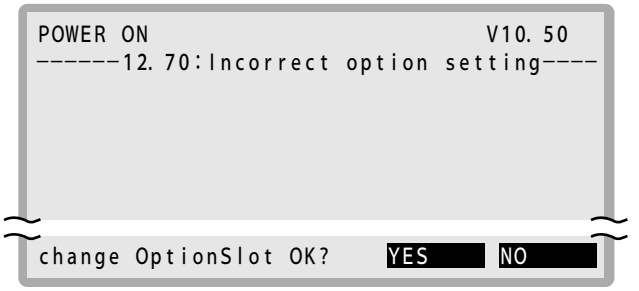
TIP

- The initial value of the "Board condition" parameters of the EtherNet/IP module is "INVALID".
- If the controller does not operate properly due to a memory error, etc., load the data saved in section 2.1. For details about how to load the data, refer to the user's manual for controller.

NOTE

For details about how to load the support software VIP+, refer to the user's manual for VIP+.

▶ **Step 2** **Checking the option setting change**



3. Setting the EtherNet/IP compatible module

To connect the EtherNet/IP system to the EtherNet/IP system specification controller, it is necessary to set the IP address, subnet mask, and gateway parameters.
These settings are made with the RPB (programming box).
The settings are valid after the controller has been restarted.

3.1 Making the EtherNet/IP module valid

To use the EtherNet/IP module, it is necessary to set the EtherNet/IP board valid using the parameters.

Step 1 ***Select "OP.BRD".***

Select "SYSTEM>PARAM>OP.BRD".
The option board list screen will appear.

Step 2 ***Select an EtherNet IP module you want to make valid.***

Move the cursor to "ENet_IP" and press the F1 *(SELECT) key.*

The Ethernet/IP module setting screen will appear.

24204-S6-00

Step 3 ***Make the Ethernet/IP module valid.***

Press the ▼ *key (cursor down key) to move the cursor to "1. Board condition" and press the* F1 *(EDIT) key.*

Press the F2 *(VALID) key. The controller will recognize the EtherNet/IP module.*

24205-S6-00

TIP

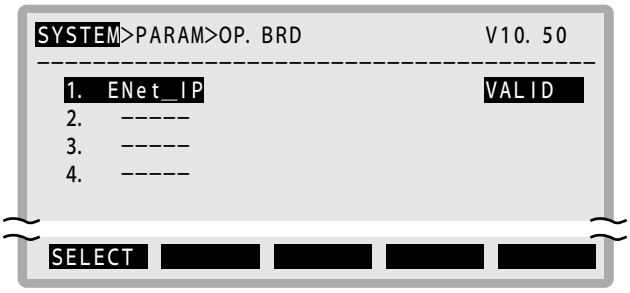
If you press the F1 (INVALID) key, the controller does not recognize the EtherNet/IP module.

Step 4 ***Exit the setting.***

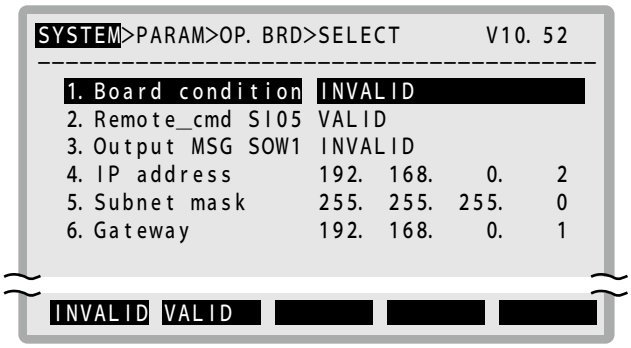
Press the ESC *key to exit the setting.*

Subsequently, to set other item, use the cursor up or down key to select a desired parameter you want to set.

Step 2 Selecting EtherNet/IP module



Step 3 Making EtherNet/IP module valid


SYSTEM>PARAM>OP. BRD>SELECT V10. 52
1. Board condition INVALID
2. Remote_cmd SI05 VALID
3. Output MSG SOW1 INVALID
4. IP address 192. 168. 0. 2
5. Subnet mask 255. 255. 255. 0
6. Gateway 192. 168. 0. 1
INVALID VALID

CAUTION

- **The "Board condition" you have changed becomes valid after the controller is restarted.**
- **This parameter has been set at "INVALID" at shipment.**

NOTE

- When using the controller without connecting to an EtherNet/IP, set the "Board condition" parameter to "INVALID".
- When the "Board condition" parameter is set to "INVALID", the dedicated input/output of the STD.DIO connector becomes enabled.
- When the "Board condition" parameter is set to "VALID", the dedicated input (except DI11 for RCX240) of the STD.DIO connector becomes disabled. However, the emergency stop input signal (DI00), interlock input (DI11), and service mode input (DI02) stay enabled.

3.2 Setting the "Remote_cmd SI05" function

This section describes how to make the remote command and I/O command function valid that uses the word information and bit information.
When the "Remote_cmd SI05" function is valid, the remote command and I/O command can be used.
Conversely, when the "Remote_cmd SI05" function is invalid, the remote command and I/O command cannot be used.
This "Remote_cmd SI05" function and "3. Output MSG SOW1" cannot be set valid at the same time.

Step 1 ***Select "OP.BRD".***

Select "SYSTEM>PARAM>OP.BRD".
The option board list screen will appear.

Step 2 ***Select an EtherNet IP module you want to make valid.***

Move the cursor to "ENet_IP" and press the F1 *(SELECT) key.*

The Ethernet/IP module setting screen will appear.

24206-S6-00

Step 3 ***Select "2. Remote_cmd SI05".***

Use the ▼ *key (cursor down key) to move the cursor to "2. Remote_cmd SI05" and press the* F1 *(EDIT) key.*

"2. Remote_cmd SI05" is then selected.

24207-S6-00

Step 4 ***Select "2. Remote_cmd SI05" valid or invalid.***

Press the F2 *(VALID) key to make the remote command and I/O command function valid.*

Press the F1 *(INVALID) key to make the remote command and I/O command function invalid.*

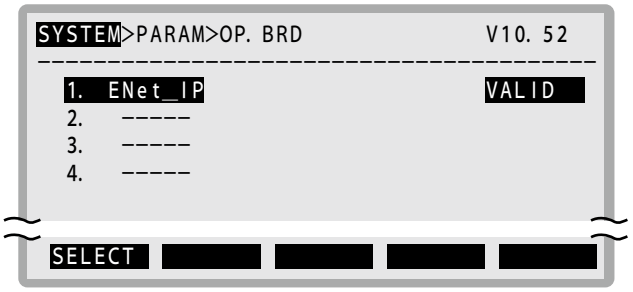
24208-S6-00

Step 5 ***Exit the setting.***

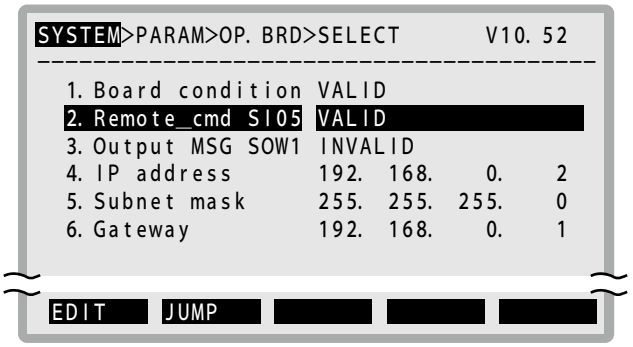
Press the ESC *key to exit the setting.*

Subsequently, to set other item, use the cursor up or down key to select a desired parameter you want to set.

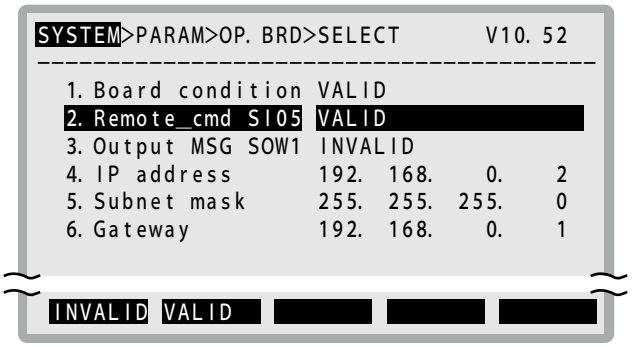
Step 2 **Selecting EtherNet/IP module**



Step 3 **Selecting "Remote_cmd SI05"**


SYSTEM>PARAM>OP. BRD>SELECT V10. 52
1. Board condition VALID
2. Remote_cmd SI05 VALID
3. Output MSG SOW1 INVALID
4. IP address 192. 168. 0. 2
5. Subnet mask 255. 255. 255. 0
6. Gateway 192. 168. 0. 1
EDIT JUMP

Step 4 **Selecting "Remote_cmd SI05" valid/invalid**


SYSTEM>PARAM>OP. BRD>SELECT V10. 52
1. Board condition VALID
2. Remote_cmd SI05 VALID
3. Output MSG SOW1 INVALID
4. IP address 192. 168. 0. 2
5. Subnet mask 255. 255. 255. 0
6. Gateway 192. 168. 0. 1
INVALID VALID

CAUTION

- **This parameter is compatible with the RCX240 software version 10.52 or higher.**
- **This parameter has been set at "VALID" at shipment.**

NOTE

- For details about remote commands and I/O commands, refer to the sections, "Remote command guide" and "I/O command guide", in this manual.
- When the "Remote_cmd SI05" function is made invalid, the "Output MSG SOW1" function cannot be used. Conversely, when the "Output MSG SOW1" function is made valid, the "Remote_cmd SI05" function cannot be used.

3.3 Setting the "Output MSG SOW1" function

This section describes how to make the function valid or invalid that outputs the message number to be displayed on the RPB to the word information SOW (1).
When the "Output MSG SOW1" function is valid, the message number to be displayed on the RPB is output to the SOW (1).
Conversely, when the "Output MSG SOW1" function is invalid, the message number to be displayed on the RPB is not output to the SOW (1).
This "Output MSG SOW1" function and "2. Remote_cmd SI05" cannot be set valid at the same time.

Step 1 ***Select "OP.BRD".***

Select "SYSTEM>PARAM>OP.BRD".
The option board list screen will appear.

Step 2 ***Select an EtherNet IP module you want to make valid.***

Move the cursor to "ENet_IP" and press the F1 *(SELECT) key.*

The Ethernet/IP module setting screen will appear.

24209-S6-00

Step 3 ***Select "3. Output MSG SOW1".***

Use the ▼ *key (cursor down key) to move the cursor to "3. Output MSG SOW1" and press the* F1 *(EDIT) key.*

"3. Output MSG SOW1" is then selected.

24210-S6-00

Step 4 ***Select "3. Output MSG SOW1" valid or invalid.***

Press the F2 *(VALID) key to make the "Output MSG SOW1" function valid.*

Press the F1 *(INVALID) key to make the "Output MSG SOW1" function invalid.*

24211-S6-00

Step 5 ***Exit the setting.***

Press the ESC *key to exit the setting.*

Subsequently, to set other item, use the cursor up or down key to select a desired parameter you want to set.

Step 2 Selecting EtherNet/IP module

```
SYSTEM>PARAM>OP. BRD                    V10. 52
-----------------------------------------------
  1.  ENet_IP                           VALID
  2.  -----
  3.  -----
  4.  -----

SELECT
```

Step 3 Selecting “Output MSG SOW1”

```
SYSTEM>PARAM>OP. BRD>SELECT             V10. 52
-----------------------------------------------
  1. Board condition VALID
  2. Remote_cmd SI05 VALID
  3. Output MSG SWW1 INVALID
  4. IP address      192.  168.    0.    2
  5. Subnet mask     255.  255.  255.    0
  6. Gateway         192.  168.    0.    1

EDIT  JUMP
```

Step 4 Selecting “Output MSG SOW1” valid/invalid

```
SYSTEM>PARAM>OP. BRD>SELECT             V10. 52
-----------------------------------------------
  1. Board condition VALID
  2. Remote_cmd SI05 VALID
  3. Output MSG SWW1 INVALID
  4. IP address      192.  168.    0.    2
  5. Subnet mask     255.  255.  255.    0
  6. Gateway         192.  168.    0.    1

INVALID  VALID
```

CAUTION

- **This parameter is compatible with the RCX240 software version 10.52 or higher.**
- **This parameter has been set at "INVALID" at shipment.**

NOTE

- For details about codes output by the "Output MSG SOW1" function, refer to the error messages stated in the user’s manual for controller.
- When the "Remote_cmd SI05" function is made invalid, the "Output MSG SOW1" function cannot be used. Conversely, when the "Output MSG SOW1" function is made valid, the "Remote_cmd SI05" function cannot be used.

3.4 Setting the IP address

Set the IP address.
The IP address is a unique number assigned to each unit that identifies multiple units connected to the network. So, the IP address needs to be set and controlled so that it does not overlap with that of other unit.

Step 1 Select "OP.BRD".

Select "SYSTEM>PARAM>OP.BRD".
The option board list screen will appear.

Step 2 Select an EtherNet IP module you want to make valid.

Move the cursor to "ENet_IP" and press the F1 *(SELECT) key.*

The Ethernet/IP module setting screen will appear.

24212-S6-00

Step 3 Select "4. IP address".

Use the ▼ *key (cursor down key) to move the cursor to "4. IP address" and press the* F1 *(EDIT) key.*

The IP address entry screen will appear.

24213-S6-00

Step 4 Enter an IP address.

Enter a desired IP address with the numeric keys (0 to 9) and period (.) key.
After you have entered the IP address, press the ENTER *key.*

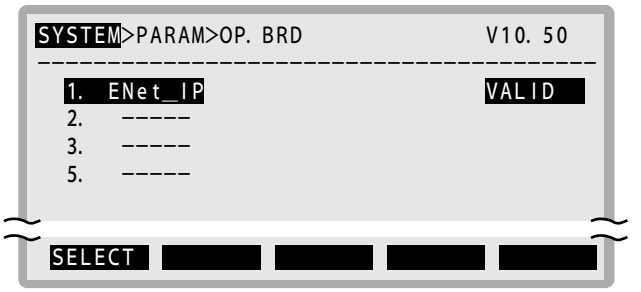
24214-S6-00

Step 5 Exit the setting.

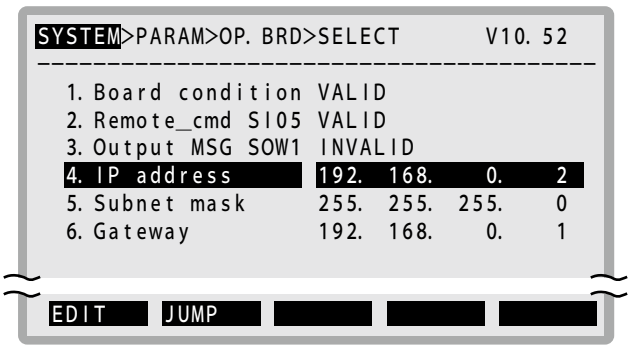
Press the ESC *key to exit the setting.*

Subsequently, to set other item, use the ▲ *or* ▼ *key (cursor up or down key) to select a desired parameter you want to set.*

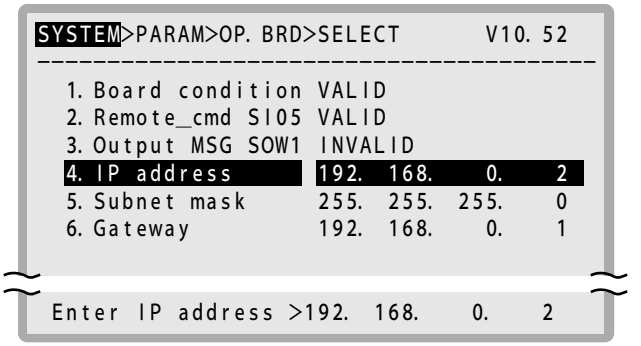
Step 2 Selecting EtherNet/IP module



Step 3 Selecting "4. IP address"


SYSTEM>PARAM>OP. BRD>SELECT V10. 52
1. Board condition VALID
2. Remote_cmd SI05 VALID
3. Output MSG SOW1 INVALID
4. IP address 192. 168. 0. 2
5. Subnet mask 255. 255. 255. 0
6. Gateway 192. 168. 0. 1
EDIT JUMP

Step 4 Entering IP address


SYSTEM>PARAM>OP. BRD>SELECT V10. 52
1. Board condition VALID
2. Remote_cmd SI05 VALID
3. Output MSG SOW1 INVALID
4. IP address 192. 168. 0. 2
5. Subnet mask 255. 255. 255. 0
6. Gateway 192. 168. 0. 1
Enter IP address >192. 168. 0. 2

3.5 Setting the subnet mask

Select the subnet mask.
The subnet mask is used to divide the network into small units.

Step 1 ***Select "OP.BRD".***

Select "SYSTEM>PARAM>OP.BRD".
The option board list screen will appear.

Step 2 ***Select an EtherNet IP module you want to make valid.***

Move the cursor to "ENet_IP" and press the F1 *(SELECT) key.*

The Ethernet/IP module setting screen will appear.

24215-S6-00

Step 3 ***Select "5. Subnet mask".***

Use the ▼ *key (cursor down key) to move the cursor to "5. Subnet mask" and press the* F1 *(EDIT) key.*

The subnet mask entry screen will appear.

24216-S6-00

Step 4 ***Enter a subnet mask.***

Enter a desired subnet mask with the numeric keys (0 to 9) and period (.) key. After you have entered the subnet mask, press the ENTER *key.*

24217-S6-00

Step 5 ***Exit the setting.***

Press the ESC *key to exit the setting.*

Subsequently, to set other item, use the ▲ *or* ▼ *key (cursor up or down key) to select a desired parameter you want to set.*

Step 2 Selecting EtherNet/IP module

```
SYSTEM>PARAM>OP. BRD                    V10. 50
-----------------------------------------------
  1.  ENet_IP                           VALID
  2.  -----
  3.  -----
  4.  -----

SELECT
```

Step 3 Selecting “5. Subnet mask”

```
SYSTEM>PARAM>OP. BRD>SELECT             V10. 52
-----------------------------------------------
  1. Board condition VALID
  2. Remote_cmd SI05 VALID
  3. Output MSG SOW1 INVALID
  4. IP address      192.  168.    0.    2
  5. Subnet mask     155.  255.  255.    0
  6. Gateway         192.  168.    0.    1

EDIT  JUMP
```

Step 4 Entering subnet mask

```
SYSTEM>PARAM>OP. BRD>SELECT             V10. 52
-----------------------------------------------
  1. Board condition VALID
  2. Remote_cmd SI05 VALID
  3. Output MSG SOW1 INVALID
  4. IP address      192.  168.    0.    2
  5. Subnet mask     155.  255.  255.    0
  6. Gateway         192.  168.    0.    1

Enter subnet mask >255.  255.  255.    0
```

CAUTION

The IP address, subnet mask, and gateway you have changed become valid after the controller is restarted. When connecting the robot controller to an existing network, be sure to check with the network administrator for the IP address, subnet mask, and gateway settings.

NOTE

The IP address is separated into network address and host address sections. The network address section is extracted from the IP address by AND processing with the subnet mask. The remaining portion is the host address section. Devices belonging to the same network must all be set to have the same network address. The host address, however, should be different for every device and set so that no two devices have the same number. The first and the last host address numbers are reserved for the system. So, be sure not to set these as the IP address.
When the IP address for example is 192.168.0.10 and the subnet mask is 255.255.255.0, the network address section is found to be 192.168.0 and the host address section to be 10 by means of AND processing with the subnet mask.
In this case, the network address section of all other devices belonging to that network must all be 192.168.0.
The host address section of those other devices on the other hand, must be set to a number other than 10. The number 0 and 255 are reserved, so do not use them for setting the host address.
So, when a device having an IP address of 192.168.0.10 and a subnet mask of 255.255.255.0 belongs to a particular network and you want to add another device to that network, then you would assign IP addresses from among 192.168.0.1 to 192.168.0.9 and 192.168.0.11 to 192.168.0.254.

NOTE

The EtherNet/IP module for RCX240 series is not usable with IP address auto acquisition functions such as DHCP and BOOTP. In this case, you must set the IP address manually.

3.6 Setting the gateway

Set the gateway. Actually, specify the IP address of the router.
The router is a device that relays the information from a certain network to other network when there are multiple networks.

Step 1 Select "OP.BRD".

Select "SYSTEM>PARAM>OP.BRD".
The option board list screen will appear.

Step 2 Select an EtherNet IP module you want to make valid.

Move the cursor to "ENet_IP" and press the F1 *(SELECT) key.*

The Ethernet/IP module setting screen will appear.

24218-S6-00

Step 3 Select "6. Gateway".

Use the ▼ *key (cursor down key) to move the cursor to "6. Gateway" and press the* F1 *(EDIT) key.*

The gateway IP address entry screen will appear.

24219-S6-00

Step 4 Enter a gateway IP address.

Enter a desired gateway IP address with the numeric keys (0 to 9) and period (.) key. After you have entered the gateway IP address, press the ENTER *key.*

24220-S6-00

Step 5 Exit the setting.

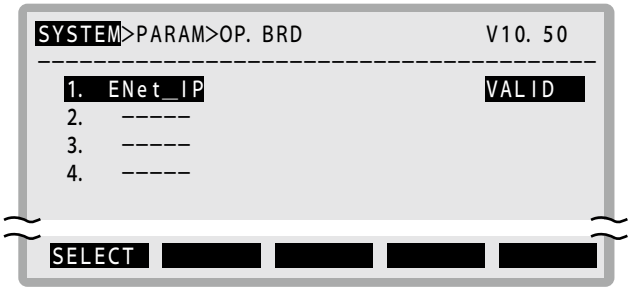
Press the ESC *key to exit the setting.*

Subsequently, to set other item, use the ▲ *or* ▼ *key (cursor up or down key) to select a desired parameter you want to set.*

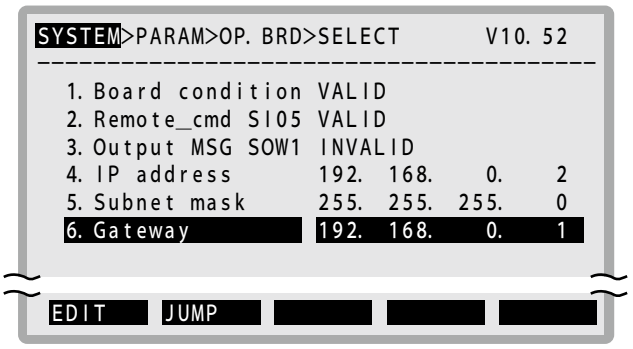
CAUTION

Any appropriate gateway address can be used as long as the network is not connected to other networks. (However, use an IP address that has not yet been assigned to other devices.)
When connecting the robot controller to an already existing network, always check with the network administrator before making IP address, subnet mask, and gateway settings.
The EtherNet/IP module for RCX240 series uses a private address as the IP address default setting. This default value cannot be used as it is on the Internet. So, when connecting to the Internet, always be sure to change the IP address of the robot controller to a global address.

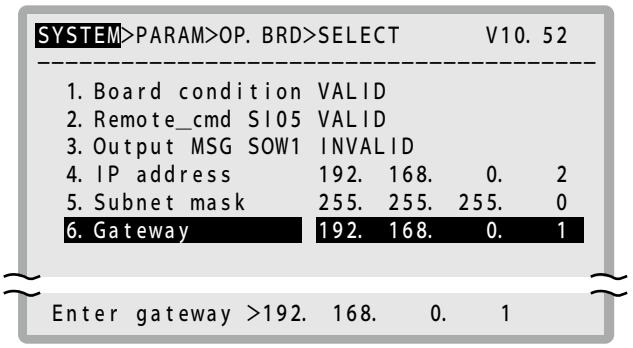
Step 2 Selecting EtherNet/IP module



Step 3 Selecting "6. Gateway"


SYSTEM>PARAM>OP. BRD>SELECT V10. 52
1. Board condition VALID
2. Remote_cmd SI05 VALID
3. Output MSG SOW1 INVALID
4. IP address 192. 168. 0. 2
5. Subnet mask 255. 255. 255. 0
6. Gateway 192. 168. 0. 1
EDIT JUMP

Step 4 Entering gateway


SYSTEM>PARAM>OP. BRD>SELECT V10. 52
1. Board condition VALID
2. Remote_cmd SI05 VALID
3. Output MSG SOW1 INVALID
4. IP address 192. 168. 0. 2
5. Subnet mask 255. 255. 255. 0
6. Gateway 192. 168. 0. 1
Enter gateway >192. 168. 0. 1

4. Noise measures

Two ferrite cores must be mounted on the shielded LAN cable when connecting to the EtherNet/IP system.

4.1 LAN cable

The EtherNet/IP is connected in a wide zone, from the enterprise zone to the manufacturing zone. So, an appropriate LAN cable that prevents noise from entering its inside must be used.

Conditions: • CAT5E grade or higher
• Twist-pair
• Dual shielded

Recommended cables: NWSMC5E-SON-S2SB-SB-*** (Straight cable) (Manufacturer: MiSUMi)
NWSMC5E-SON-C2SB-SB-*** (Cross cable) (Manufacturer: MiSUMi)
(* shows the cable length. A desired cable length can be specified at intervals of 0.1m in a range of 0.5 to 100m.)

4.2 Mounting the ferrite core

Mount one ferrite core at both ends of the LAN cable.

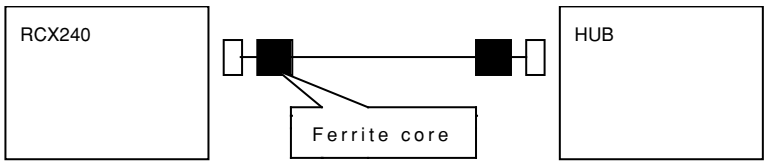
WARNING
COMPLETELY SHUT DOWN THE POWER SUPPLY TO THE INPUT POWER CABLE BEFORE STARTING THIS WORK.

■ Procedure

Step 1 ***Mount the ferrite core.***
Mount the ferrite core at both ends of the LAN cable as shown in the figure below. At this time, place the ferrite core as close to the robot controller and HUB as possible.

Step 2 ***Secure the ferrite core with a cable tie, etc.***
Secure the mounted ferrite core with a cable tie, etc.

Mounting ferrite core



23201-S6-00

5. Connecting to the EtherNet/IP system

5.1 Connecting the LAN cable

WARNING

BEFORE CONNECTING THE CABLE, COMPLETELY SHUT DOWN THE POWER SUPPLIED TO THE ROBOT CONTROLLER.

Insert the modular jack of the LAN cable recommended in section 4.1, LAN cable, into the modular connector of the controller until a click sounds. In the same manner, connect the modular jack into the modular connector of the hub.

CAUTION

In the EtherNet/IP, it is recommended to use a hub that connects the chassis of the LAN connector to the PE. YAMAHA also conducts the functional check with the hub that connects the chassis to the PE.

CAUTION

The maximum length of the cable between the hub and controller is 100 m.
When connecting the LAN cable, be sure to thoroughly read the user's manuals for mating units, such as personal computer and sequencer, and peripheral units, such as hub.

MEMO

It is recommended to connect the mating unit with the straight cable through the hub. It is also possible to use the cross-cable to directly connect the mating unit without use of the hub. At this time, however, the communication with the mating unit may not be performed correctly depending on the type of the LAN adaptor.

Chapter 3 Communication

Contents

1. State when the robot controller power is turned ON 3-1
2. Communication with the master module 3-2
   - 2.1 Receiving data 3-2
   - 2.2 Transmitting data 3-4
3. Direct connection by emulated serialization on parallel DIO 3-5
   - 3.1 Emulated serialization setting on parallel DIO 3-5
4. Referring to the communication data 3-8
   - 4.1 Referring to the data from the programming box 3-8

1. State when the robot controller power is turned ON

The EtherNet/IP system specification robot controller always starts the operation in the servo OFF state after the power has been turned ON.

1. When the connection to the EtherNet/IP system is correctly established.

The following conditions must be satisfied to correctly connect to the EtherNet/IP system.

- The EtherNet/IP system cable must be physically connected.
- The IP address, subnet mask, and gateway must be set correctly.
- The master module is operating correctly.

When connected to the EtherNet/IP system correctly, the LEDs on the EtherNet/IP compatible module show the normal state.
At this time, the emergency stop signal and interlock signal in the EtherNet/IP system become valid, so both signals need to be turned ON in the connection process.
The emergency stop terminal in the SAFETY connector is always valid.
In the RCX240, the interlock signal of STD.DIO is valid unless the external 24V monitor control of the system parameters is set invalid.
When the SAFE mode is enabled, the SERVICE mode input signal is made valid with SI(02) in the EtherNet/IP system. In the RCX240, when the SAFE mode is enabled and the external 24V monitor control of the system parameters is not set invalid, the SERVICE mode input signal is made valid with DI(02) of the SAFETY connector.

2. When the connection to the EtherNet/IP system is incorrectly established.

If connected to the EtherNet/IP system incorrectly, the following may be the cause.

- The EtherNet/IP system cable is not physically connected.
- The IP address, subnet mask, or gateway is not set correctly.
- The master module is not operating correctly.

If connected to the EtherNet/IP system incorrectly, the LEDs on the EtherNet/IP compatible module show the error state. This also occurs when the master module is not operating correctly.
At this time, the emergency stop signal and interlock signal in the EtherNet/IP system turn off. So, the robot controller cannot be operated individually.
To operate the robot controller individually, change the "Board condition" parameter of the EtherNet/IP module to "INVALID".
The emergency stop signal terminal in the SAFETY connector is always valid.
In the RCX240, the interlock signal of STD.DIO is valid unless the external 24V monitor control of the system parameters is set invalid.
In the RCX240, when the SAFE mode is enabled and the external 24V monitor control of the system parameters is not set invalid, the SERVICE mode input signal is made valid with DI (02) in the SAFETY connector.
When the SAFE mode is enabled, the SERVICE mode input signal in the EtherNet/IP system cannot be set invalid. So, change the SERVICE mode setting of the system parameters for operation. Always change this parameter with great care.

MEMO
For details about LED indications, see Chapter 4 of this guide.

2. Communication with the master module

This section describes the communication with the master module using the robot program when connected to the EtherNet/IP system correctly.

2.1 Receiving data

The data in the output area of the master module is read via the serial input ports of the robot controller. The following shows the correspondence between the output area of the master module and the serial input port of the robot controller.

Address of master module output area	Serial input port No. of robot controller		Address of master module output area	Serial input port No. of robot controller
n		SIW(0)	n+32	SI0(7~0)
n+2		SIW(1)	n+33	SI1(7~0)
n+4	SID(2)	SIW(2)	n+34	SI2(7~0)
n+6		SIW(3)	n+35	SI3(7~0)
n+8	SID(4)	SIW(4)	n+36	SI4(7~0)
n+10		SIW(5)	n+37	SI5(7~0)
n+12	SID(6)	SIW(6)	n+38	SI6(7~0)
n+14		SIW(7)	n+39	SI7(7~0)
n+16	SID(8)	SIW(8)	n+40	SI10(7~0)
n+18		SIW(9)	n+41	SI11(7~0)
n+20	SID(10)	SIW(10)	n+42	SI12(7~0)
n+22		SIW(11)	n+43	SI13(7~0)
n+24	SID(12)	SIW(12)	n+44	SI14(7~0)
n+26		SIW(13)	n+45	SI15(7~0)
n+28	SID(14)	SIW(14)		
n+30		SIW(15)		

n : Start address of the output area assigned to the master module

CAUTION

Before communicating with the master module, be sure to check the setting while referring to the user's manual for PLC.

When reading the bit information from the output area of the master module with the robot controller, write the following commands in the robot program in the same manner as the DI input port.

WAIT command
Assignment statement

Example: To wait for bit 0 of the address (n+34) to turn ON.
WAIT SI (20) = 1 ..... The robot program will wait for SI(20) to turn ON.

Example: To read the address (n+34)0 to (n+34)7 data into variable A.
A = SI2 () ..... The SI2() data will be converted into a decimal value and assigned to variable A. If SI2() is 7Fh, variable A will be 127.

NOTE

The SI statement in the robot language can be defined from SI0() to SI27(), but the EtherNet/IP compatible module accepts from SI0() to SI15().

When reading the word information from the output area of the master module with the robot controller, create the robot program using the assignment statement.

Example: To read the address (n+4) word data into variable B.
B = SIW (2) ..... The SIW(2) data will be assigned to variable B as a decimal value. If SIW(2) is 01FFh, variable B will be 511.

Example: To read the address (n+4) and (n+6) double word data into variable C.
C = SID (2)..... The SIW(2) and SIW(3) data will be assigned to variable C as a decimal value. If SIW(2) is 0010h and SIW(3) is 0001h, variable C will be 65552.

NOTE

The word data written with SIW(n) has the uncoded little endian format.
The double word data written with SID(n) has the coded little endian format.

2.2 Transmitting data

The serial output port data of the robot controller is transmitted to the input area of the master module. The correspondence between the serial output port of the robot controller and the input area of the master module is shown below.

Address of master module input area	Serial output port No. of robot controller		Address of master module input area	Serial output port No. of robot controller
m		SOW(0)	m+32	SO0(7~0)
m+2		SOW(1)	m+33	SO1(7~0)
m+4	SOD(2)	SOW(2)	m+34	SO2(7~0)
m+6		SOW(3)	m+35	SO3(7~0)
m+8	SOD(4)	SOW(4)	m+36	SO4(7~0)
m+10		SOW(5)	m+37	SO5(7~0)
m+12	SOD(6)	SOW(6)	m+38	SO6(7~0)
m+14		SOW(7)	m+39	SO7(7~0)
m+16	SOD(8)	SOW(8)	m+40	SO10(7~0)
m+18		SOW(9)	m+41	SO11(7~0)
m+20	SOD(10)	SOW(10)	m+42	SO12(7~0)
m+22		SOW(11)	m+43	SO13(7~0)
m+24	SOD(12)	SOW(12)	m+44	SO14(7~0)
m+26		SOW(13)	m+45	SO15(7~0)
m+28	SOD(14)	SOW(14)		
m+30		SOW(15)		

m : Start address of the input area assigned to the master module

CAUTION

Before communicating with the master module, be sure to check the setting while referring to the user's manual for PLC.

When writing the bit information of the robot controller to the input area of the master module, write the following commands in the robot program in the same manner as the DO input port.

SET/RESET command
Assignment statement
OUT command

Example: To turn the address (m+34)0 ON.
SET SO (20) or SO (20) = 1 ... SO (20) will turn ON.

Example: To write the variable A data to addresses (m+34)0 to (m+34)7.
SO2 () = A ..... The variable A data will be converted into a binary value and assigned to SO2(). If variable A is 127, SO2() will be 7Fh.

NOTE

The SO statement in the robot language can be defined from SO2() to SO27(), but the EtherNet/IP compatible module accepts from SO2() to SO15().

When writing the word information of the robot controller to the input area of the master module, create the robot program using the assignment statement.

Example: To write 512 into addresses (m+4) as word data.
SOW (2) = 512..... 512 is assigned to SOW(2), and then SOW(2) becomes 0200h.

Example: To write 69905 to addresses (m+4) and (m+6) as double word data.
SOD (2) = 69905..... 69905 is assigned to SOD(2), and then SOW(2) becomes 1111h and SOW(3) becomes 0001h.

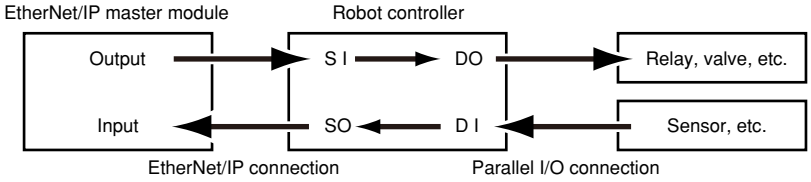
NOTE

The word data written with SOW(n) has the uncoded little endian format.
The double word data written with SOD(n) has the coded little endian format.

3. Direct connection by emulated serialization on parallel DIO

The robot controller's parallel input data can be transferred to the serial output data regardless of the robot program. Likewise, the robot controller's serial input data can be transferred to the parallel output data.
By using this function, a sensor or relay connected to the parallel I/O of the robot controller can be used like a device connected to the EtherNet/IP master module.

Direct connection by emulated serialization



23301-S6-00

NOTE

When the directly connected and set output port is used with the program, the bit information may not become the intended value. Do not use the directly connected and set output port with the program.

3.1 Emulated serialization setting on parallel DIO

The relation of the parallel port and serial port that can be connected is shown below.

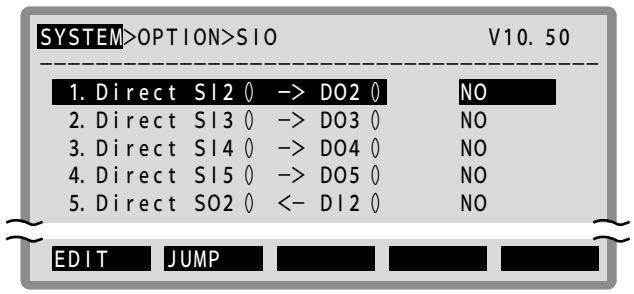
Input device such as sensor		Output device such as valve	
DI port → SO port		DO port ← SI port	
DI2()	SO2()	DO2()	SI2()
DI3()	SO3()	DO3()	SI3()
DI4()	SO4()	DO4()	SI4()
DI5()	SO5()	DO5()	SI5()

Operation

Press the F3 (SIO) key in the "SYSTEM >OPTION" mode.

The setting screen will appear that is necessary to make the emulated serialization setting on parallel DIO.

SIO setting screen


SYSTEM>OPTION>SIO V10.50
1. Direct SI2() -> DO2() NO
2. Direct SI3() -> DO3() NO
3. Direct SI4() -> DO4() NO
4. Direct SI5() -> DO5() NO
5. Direct SO2() <- DI2() NO
EDIT
JUMP

24301-S6-00

Valid keys and submenu functions in this mode are as follows.

Valid keys	Menu	Function
▲ / ▼		Selects SIO parameters.
F1	EDIT	Sets SIO parameters.
F2	JUMP	Jumps to specified SIO parameter.

NOTE

When the port specified by SIO is identical with the port used by the program, the output results might be inaccurate.

1. Direct connection from SI n ( ) to DO n ( )

Serial port input can be directly connected to the parallel port output. The relation of the parallel port and serial port that can be connected is as follows.

Output device such as sensor	
DO port ← SI port	
DO2()	SI2()
DO3()	SI3()
DO4()	SI4()
DO5()	SI5()

NOTE

When the port specified by SIO is identical with the port used by the program, the output results might be inaccurate.

■ Operation

Step 1 ***Select "SIO".***

Select "SYSTEM > OPTION > SIO".
The SIO setting screen will appear.

Step 2 ***Select a SI port.***

Use the ▲ or ▼ key (cursor up or down key) to move the cursor to a desired SI port (from "1. Direct SI2() -> DO2()" to "4. Direct SI5() -> DO5()") and press the F1 (EDIT) key.

Step 3 ***Set the direct connection.***

Press the F1 (SET) key to make the direct connection valid.

Press the F2 (NO) key to make the direct connection invalid.

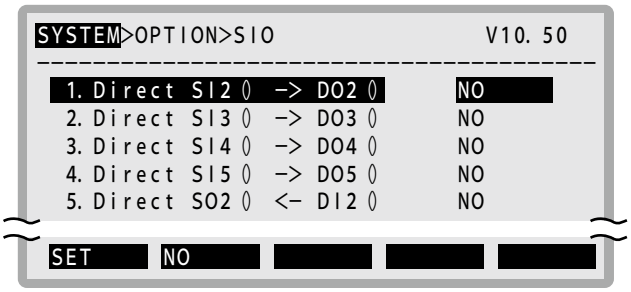
24302-S6-00

Step 4 ***Exit the setting.***

Press the ESC key to exit the setting.

Subsequently, to set other item, use the ▲ or ▼ key (cursor up or down key) to select other SI port.

▶ **Step 3** Setting SI port direct connection


SYSTEM>OPTION>SIO V10. 50
1. Direct SI2 () -> DO2 () NO
2. Direct SI3 () -> DO3 () NO
3. Direct SI4 () -> DO4 () NO
4. Direct SI5 () -> DO5 () NO
5. Direct SO2 () <- DI2 () NO
SET
NO

2. Direct connection from DI n ( ) to SDO n ( )

Parallel port input can be directly connected to the serial port output. The relation of the parallel port and serial port that can be connected is as follows.

Input device such as valve	
DI port → SO port	
DI2()	SO2()
DI3()	SO3()
DI4()	SO4()
DI5()	SO5()

NOTE

When the port specified by SIO is identical with the port used by the program, the output results might be inaccurate.

■ Operation

Step 1 ***Select "SIO".***

Select "SYSTEM > OPTION > SIO".
The SIO setting screen will appear.

Step 2 ***Select a DI port.***

Use the ▲ or ▼ key (cursor up or down key) to move the cursor to a desired DI port (from "5. Direct SO2() <- DI2()" to "8. Direct SO5() <- DI5()") and press the F1 (EDIT) key.

Step 3 ***Set the direct connection.***

Press the F1 (SET) key to make the direct connection valid.

Press the F2 (NO) key to make the direct connection invalid.

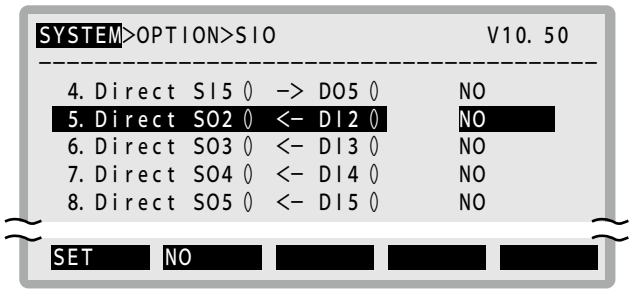
24303-S6-00

Step 4 ***Exit the setting.***

Press the ESC key to exit the setting.

Subsequently, to set other item, use the ▲ or ▼ key (cursor up or down key) to select other DI port.

▶ **Step 3** Setting DI port direct connection


SYSTEM>OPTION>SIO V10. 50
4. Direct SI5 () -> DO5 () NO
5. Direct SO2 () <- DI2 () NO
6. Direct SO3 () <- DI3 () NO
7. Direct SO4 () <- DI4 () NO
8. Direct SO5 () <- DI5 () NO
SET
NO

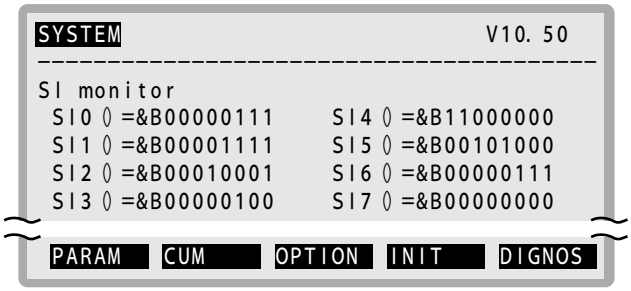
4. Referring to the communication data

The ON/OFF information exchanged with the master module can be referred to with the RPB. Note that the RPB display update interval is longer than the EtherNet/IP data update interval. So, if the ON/OFF interval is short, accurate information may not be displayed.

4.1 Referring to the data from the programming box

The data exchanged with the master module can be referred to with the RPB. The reference unit is the robot controller input/output port No.

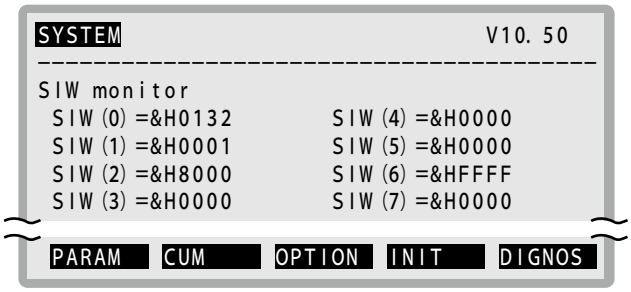
Communication screen with RPB



24304-S6-00

* &Bxxxxxxx corresponds to the 0th bit to 7th bit from the right to the left.

Reference screen from RPB



24305-S6-00

* &Hxxxx expresses a hexadecimal value.

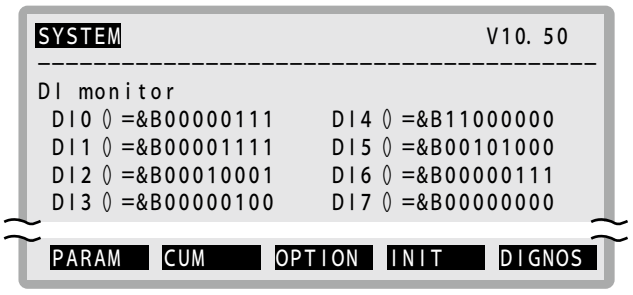
■ Operation

Step 1 ***Display the DI monitor.***

Press the DISPLAY *key on the RPB. The DI monitor screen will appear.*

24306-S6-00

▶ **Step 1** **DI monitor**



Step 2 Display the input/output port status.

Press the DISPLAY *key on the RPB until the SI input ports 0 to 7 are displayed.*
(See the figure shown on the right.)

24207-S6-00

The input/output port status is displayed in the order shown below each time the DISPLAY *key is pressed from this status.*

SI input ports 10 to 15
↓
Unused.
↓
SO input ports 0 to 7
↓
SO input ports 10 to 15
↓
Unused.
↓
SIW input ports 0 to 7
↓
SIW input ports 8 to 15
↓
SOW output ports 0 to 7
↓
SOW output ports 8 to 15

Step 3 Exit the DI monitor.

After you have checked the input/output port status, press the ESC *key to exit the DI monitor.*

Step 2 Displaying SI input ports 0 to 7

```
SYSTEM                                  V10.50
-----------------------------------------------
SI monitor
 SI0 () =&B00000111     SI4 () =&B11000000
 SI1 () =&B00001111     SI5 () =&B00101000
 SI2 () =&B00010001     SI6 () =&B00000111
 SI3 () =&B00000100     SI7 () =&B00000000

PARAM  CUM  OPTION  INIT  DIGNOS
```

Chapter 4 Troubleshooting

Contents

1. Check items before starting up the EtherNet/IP system 4-1
2. Meanings of LEDs on EtherNet/IP compatible module 4-2
3. Troubleshooting 4-3
   - 3.1 Robot controller front panel LED confirmation 4-3
   - 3.2 Programming box error display confirmation 4-3
   - 3.3 EtherNet/IP compatible module LED confirmation 4-4
   - 3.4 Confirmation from master module 4-4
4. Error messages relating to EtherNet/IP 4-5

1. Check items before starting up the EtherNet/IP system

Check the following items before starting up the EtherNet/IP system.

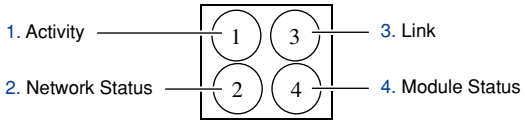
	Check item	Check
1	Is the EtherNet/IP compatible module connected securely? (Refer to section 1 in Chapter 2.)	
2	Is the robot controller set to the EtherNet/IP system specifications? (Refer to section 2 in Chapter 2.)	
3	Are the IP address, subnet mask, and gateway of the EtherNet/IP compatible module are set correctly? (Refer to section 3 in Chapter 2.)	
4	Are the ferrite cores connected to the power input cable to the robot controller? (Refer to section 4 in Chapter 2.)	
5	Is the EtherNet/IP system cable connected to the EtherNet/IP compatible module securely? (Refer to section 5 in Chapter 2.)	
6	Was the line test from the master module correct? (Refer to the user's manual for master module.)	

NOTE

In the RCX240, the dedicated input of STD. DIO provided on the controller will be disabled except for the interlock signal (DI11). When the external 24V monitor control of the system parameters is set invalid, the interlock signal (DI11) will also be disabled.

2. Meanings of LEDs on EtherNet/IP compatible module

Status indicators



23401-S6-00

The LEDs on the EtherNet/IP compatible module express the following statuses.
Use these for the confirmation work if an error occurs.

1. Activity : Flashing green during packet communication.

2. Network Status :

Status	Description
OFF	Power is OFF or no IP address is found.
Lit in green	Detects the online and connects other unit.
Flashing green	Detects the online, but does not connect other unit.
Lit in red	Detects serious error, such as IP address duplication.
Flashing red	Time-out occurs during connection with other unit.
Flashing green/red	Performing the self-test (only when the power is turned ON).

3. Link : Lit in green when the link is detected.

4. Module Status :

Status	Description
OFF	Power is OFF.
Lit in green	Connection with the master is correct.
Flashing green	Connection with the master is not established.
Lit in red	Recoverable minor error is detected.
Flashing red	Unrecoverable error is detected.
Flashing green/red	Performing the self-test (only when the power is turned ON).

3. Troubleshooting

3.1 Robot controller front panel LED confirmation

■ Confirmation item 1

Confirmation contents	• The "PWR" LED is OFF.
Cause	• The power is not supplied to the robot controller.
Corrective measures	• Measure the voltage at the AC power input terminal of the power connector with a multi-meter to check that the operating power voltage is supplied.

MEMO

For details about the operating power voltage for the robot controller, refer to the user's manual for robot controller.

■ Confirmation item 2

Confirmation contents	• The "ERR" LED is ON.
Cause	• The robot controller is in the emergency stop state. • A serious error has occurred in the robot controller.
Corrective measures	• Check the error message displayed on the programming box. • Take corrective measures while referring to the troubleshooting stated in the user's manual for robot controller.

MEMO

For details about error contents, refer to the user's manual for robot controller.

3.2 RPB (programming box) error display confirmation

■ Confirmation item

Confirmation contents	• Check the error message displayed on the RPB.
Cause	• An error has occurred in the robot controller.
Corrective measures	• Check the error message displayed on the RPB. • Check the error history with the RPB. At this time, check the error history with the RPB in the "SYSTEM> DIAGNOS > HISTORY" mode. • Take corrective measures while referring to the troubleshooting stated in the user's manual for robot controller.

MEMO

For details about error contents, see the user's manual for robot controller.

3.3 EtherNet/IP compatible module LED confirmation

■ Confirmation item

Confirmation contents	• Check that the LED indication on the EtherNet/IP compatible module is not "ACT: Flashing green", "Link: Lit in green", "MS: Lit in green", and "NS: Lit in green".
Cause	• An error has occurred in the EtherNet/IP system connection. • For details about LED indication meanings, refer to the table stated in "2. Meanings of LEDs on EtherNet/IP compatible module" of Chapter 4 in this guide.
Corrective measures	• Check whether the LAN cable is not connected or connected incorrectly, and whether or not the straight or cross cable does not meet the specifications. • Check whether the LAN cable is run close to the main circuit or power cable or whether or not it is bundled. • Check that the ferrite core is connected to the power supply cable of the robot controller. • Check the TCP/IP setting of the EtherNet/IP compatible module. • Check that the master module is operating correctly.

3.4 Confirmation from master module

■ Confirmation item

Confirmation contents	• Use the connection setting function or connection check function of the master module to check that the robot controller is connected to the EtherNet/IP system correctly.
Cause	• The signal has noise. • The cable is broken. • The IP address setting is incorrect.
Corrective measures	• Replace the cable. • Change the cable running route to lay it away from the noise source, such as power cable. • Check the IP address setting.

MEMO

For details about connection setting function, refer to the user's manual for master module.
Furthermore, for details about IP address and other settings, contact the system administrator.

4. Error messages relating to EtherNet/IP

This section describes error messages relating to EtherNet/IP compatible modules. For other messages, refer to the user's manual for robot controller.
If an error occurs, relevant error message appears on the message line (2nd line) of the RPB screen.

12.1 : Emg.stop on

Code : &H0C01

Meaning/Cause	a. RPB emergency stop button was pressed. b. Emergency stop terminals in the SAFETY connector are open (emergency stop state). c. RPB or terminator is not connected to the RPB connector. d. SAFETY connector is not connected. e. SI(00) is not ON. f. Error in the connection to the EtherNet/IP system.
Action	1. Release the RPB emergency stop button. 2. Close the emergency stop terminals in the SAFETY connector. 3. Connect the RPB or terminator to the RPB connector. 4. Attach the SAFETY connector. 5. Set SI(00) to ON. 6. Correct the connection to the EtherNet/IP system.

12.2 : Interlock on

Code : &H0C02

Meaning/Cause	a. Program was executed or moving of axis attempted with the interlock signal still input. b. Interlock signal turned ON during execution of program or axis movement. c. DC 24V is supplied to the STD.DIO connector and DI(11) is not turned ON while using the RCX240. d. SI(11) is not ON. e. Error in the connection to the EtherNet/IP system.
Action	1. Cancel the interlock signal, and execute the program or move the axis. 2. Set DI(11) on the STD.DIO connector to ON. (RCX240) 3. Set SI(11) to ON. 4. When not using STD.DIO, disable (invalid) the "Watch on STD.DO DC24V" parameter in the SYSTEM mode. (RCX240) 5. Correct the connection to the EtherNet/IP system.

12.51 : EtherNet/IP link error

Code : &H0C33

Meaning/Cause	a. Error in the EtherNet/IP option board.
Action	1. Contact our company.

12.70 : Incorrect option setting

Code : &H0C46

Meaning/Cause	a. Error in the DIP switch setting on the option unit. b. Mismatched option units have been installed. c. Cannot identify the installed option unit.
Action	1. Check the DIP switch settings on the option unit. 2. Install the correct option units. 3. Replace the option unit.

21.16 : System error (EherNet/IP)

Code : &H1510

Meaning/Cause	a. Error in the software.
Action	1. Contact our company.

Chapter 5 Specifications

Contents

1. Profile 5-1
2. Details of input/output signals 5-3
3. Dedicated input/output signal timing chart 5-6
   - 3.1 Servo ON and emergency stop 5-6
   - 3.2 AUTO mode changeover, program reset and program execution 5-7
   - 3.3 Stopping with program interlock 5-8
4. EtherNet/IP compatible module specifications 5-9
5. EtherNet/IP specifications 5-10

1. Profile

■ Bit input/output

Slave → Master				Master → Slave			
Address	Bit	Signal name		Address	Bit	Signal name	
m+32	0	SO(00)	Emergency stop input status output	n+32	0	SI(00)	Emergency stop input
	1	SO(01)	CPU_OK status output		1	SI(01)	Servo ON input
	2	SO(02)	Servo ON status output		2	SI(02)	Service mode input
	3	SO(03)	Alarm status output		3		Reserved.
	4		Reserved.		4		Reserved.
	5		Reserved.		5	SI(05)	I/O command execution trigger input
	6		Reserved.		6		Reserved.
	7		Reserved.		7		Reserved.
m+33	0	SO(10)	AUTO mode status output	n+33	0	SI(10)	Sequence control input
	1	SO(11)	Return-to-origin complete status output		1	SI(11)	Interlock input
	2	SO(12)	Sequence program execution status output		2	SI(12)	Start input
	3	SO(13)	Robot program execution status output		3	SI(13)	AUTO mode input
	4	SO(14)	Program reset status output		4	SI(14)	Return-to-origin input
	5		Reserved.		5	SI(15)	Program reset input
	6	SO(16)	I/O command execution judgment output		6	SI(16)	MANUAL mode input
	7	SO(17)	Output during I/O command execution		7	SI(17)	Absolute reset / Return-to-origin input *1
m+34	0~7	SO(20) ~ SO(27)	General-purpose output	n+34	0~7	SI(20) ~ SI(27)	General-purpose input
m+35	0~7	SO(30) ~ SO(37)	General-purpose output	n+35	0~7	SI(30) ~ SI(37)	General-purpose input
m+36	0~7	SO(40) ~ SO(47)	General-purpose output	n+36	0~7	SI(40) ~ SI(47)	General-purpose input
m+37	0~7	SO(50) ~ SO(57)	General-purpose output	n+37	0~7	SI(50) ~ SI(57)	General-purpose input
m+38	0~7	SO(60) ~ SO(67)	General-purpose output	n+38	0~7	SI(60) ~ SI(67)	General-purpose input
m+39	0~7	SO(70) ~ SO(77)	General-purpose output	n+39	0~7	SI(70) ~ SI(77)	General-purpose input
m+40	0~7	SO(100) ~ SO(107)	General-purpose output	n+40	0~7	SI(100) ~ SI(107)	General-purpose input
m+41	0~7	SO(110) ~ SO(117)	General-purpose output	n+41	0~7	SI(110) ~ SI(117)	General-purpose input
m+42	0~7	SO(120) ~ SO(127)	General-purpose output	n+42	0~7	SI(120) ~ SI(127)	General-purpose input
m+43	0~7	SO(130) ~ SO(137)	General-purpose output	n+43	0~7	SI(130) ~ SI(137)	General-purpose input
m+44	0~7	SO(140) ~ SO(147)	General-purpose output	n+44	0~7	SI(140) ~ SI(147)	General-purpose input
m+45	0~7	SO(150) ~ SO(157)	General-purpose output	n+45	0~7	SI(150) ~ SI(157)	General-purpose input

(continued to next page)

Slave → Master				Master → Slave			
Address	Bit	Signal name		Address	Bit	Signal name	
m+46	0~7		Reserved.	n+46	0~7		Reserved.
m+47	0~7		Reserved.	n+47	0~7		Reserved.

m : Start address of the input area assigned to the master module
n : Start address of the output area assigned to the master module

*1: Used for "absolute reset" or "absolute reset / return-to-origin" depending on the parameter (DI17 mode) setting.

■ Word input/output

Slave → Master				Master → Slave			
Address	Name			Address	Name		
m		SOW(0)	Dedicated output	n		SIW(0)	Dedicated input
m+2		SOW(1)	Dedicated output	n+2		SIW(1)	Dedicated input
m+4	SOD(2)	SOW(2)	General-purpose output	n+4	SID(2)	SIW(2)	General-purpose input
m+6		SOW(3)	General-purpose output	n+6		SIW(3)	General-purpose input
m+8	SOD(4)	SOW(4)	General-purpose output	n+8	SID(4)	SIW(4)	General-purpose input
m+10		SOW(5)	General-purpose output	n+10		SIW(5)	General-purpose input
m+12	SOD(6)	SOW(6)	General-purpose output	n+12	SID(6)	SIW(6)	General-purpose input
m+14		SOW(7)	General-purpose output	n+14		SIW(7)	General-purpose input
m+16	SOD(8)	SOW(8)	General-purpose output	n+16	SID(8)	SIW(8)	General-purpose input
m+18		SOW(9)	General-purpose output	n+18		SIW(9)	General-purpose input
m+20	SOD(10)	SOW(10)	General-purpose output	n+20	SID(10)	SIW(10)	General-purpose input
m+22		SOW(11)	General-purpose output	n+22		SIW(11)	General-purpose input
m+24	SOD(12)	SOW(12)	General-purpose output	n+24	SID(12)	SIW(12)	General-purpose input
m+26		SOW(13)	General-purpose output	n+26		SIW(13)	General-purpose input
m+28	SOD(14)	SOW(14)	General-purpose output	n+28	SID(14)	SIW(14)	General-purpose input
m+30		SOW(15)	General-purpose output	n+30		SIW(15)	General-purpose input

m : Start address of the input area assigned to the master module
n : Start address of the output area assigned to the master module

2. Details of input/output signals

■ Bit output

Address	Signal name		Description
(m+32)0	SO(00)	Emergency stop input status output	Turns ON when the robot controller is in the emergency stop state.
(m+32)1	SO(01)	CPU_OK status output	Turns ON when the robot controller is in the normal state.
(m+32)2	SO(02)	Servo ON status output	Turns ON when the motor power of the robot controller is ON.
(m+32)3	SO(03)	Alarm status output	Turns ON when the robot controller is in any of the following states. • The controller is in the serious error state. • The emergency stop input is turned OFF.
(m+33)0	SO(10)	AUTO mode status output	Turns ON when the AUTO mode is selected. Turns OFF when other mode is selected.
(m+33)1	SO(11)	Return-to-origin complete status output	Turns ON when the robot has completed the return-to-origin.
(m+33)2	SO(12)	Sequence program execution status output	Turns ON while the sequence program is being executed.
(m+33)3	SO(13)	Robot program execution status output	Turns ON while the robot program is being executed.
(m+33)4	SO(14)	Program reset status output	Turns ON when the robot program has been reset. Turns OFF when the robot program starts.
(m+33)6	SO(16)	I/O command execution judgment output	Turns OFF while the I/O command is being executed. After the I/O command has been executed, turns ON if normal and stays OFF if abnormal.
(m+33)7	SO(17)	Output during I/O command execution	Turns ON while the I/O command is being executed.
(m+34)0 ~ (m+34)7	SO(20) ~ SO(27)	General-purpose output	General-purpose output turns ON/OFF when the value is assigned to the SO port, or SET/RESET command or OUT command is executed.
~	~	~	
(m+44)0 ~ (m+44)7	SO(150) ~ SO(157)	General-purpose output	

m : Start address of the input area assigned to the master module

NOTE

When the area check output function is used, the area check outputs can be assigned to the general-purpose outputs SO(20) to SO(157).

Bit input

Address	Signal name		Description
(n+32)0	SI(00)	Emergency stop input	Turn OFF to put the controller in the emergency stop state. Keeps turned ON during normal operation.
(n+32)1	SI(01)	Servo ON input	Turn ON to cancel the emergency stop state and put the robot servomotor in the ON state. The servo ON is executed when this signal is switched from OFF to ON. It is necessary that the emergency stop input [SI(00)] is in the ON state and all emergency stop states (emergency stop terminal in the SAFETY connector, etc.) on the robot controller are cancelled.
(n+32)2	SI(02)	Service mode input	Turn OFF to put the controller in the SERVICE mode. Keeps turned ON during normal operation. (This signal is valid only when the SAFE mode is set.) (When the SAFE mode is set and the controller is in the SERVICE mode, the dedicated input may become invalid depending on the SERVICE mode setting.)
(n+32)5	SI(05)	I/O command execution trigger input	Switch this signal from OFF to ON to execute the I/O command. Note that this signal must be turned ON when the I/O command is set for the general-purpose input.
(n+33)0	SI(10)	Sequence control input	Turn ON to execute the sequence program in the robot controller. The sequence program is executed when this signal is in the ON state.
(n+33)1	SI(11)	Interlock input	Turn OFF to stop the robot program currently being executed. To execute the program, keep this signal turned ON.
(n+33)2	SI(12)	Start input	Turn ON to execute the robot program. The robot program is executed when this signal is switched from OFF to ON. It is necessary that the robot controller is in the AUTO mode.
(n+33)3	SI(13)	AUTO mode input	Turn ON to select the AUTO mode. The mode transits to the AUTO mode when this signal is switched from OFF to ON.
(n+33)4	SI(14)	Return-to-origin input	Turn ON to perform the return-to-origin of the incremental type axis or semi-absolute type axis. When this signal is switched from OFF to ON, the incremental type axis performs the return-to-origin and the semi-absolute type axis performs the absolute search operation. This signal is intended for axes whose return-to-origin method is the sensor or stroke end method. It is necessary that the robot controller is in the MANUAL mode.
(n+33)5	SI(15)	Program reset input	Turn ON to reset the robot program. The program is reset when this signal is switched from OFF to ON. It is necessary that the robot controller is in the AUTO mode.
(n+33)6	SI(16)	MANUAL mode input	Turn ON to select the MANUAL mode. The mode transits to the MANUAL mode when this signal is switched from OFF to ON.
(n+33)7	SI(17)	Absolute reset / Return-to-origin input	Used for "absolute reset" or "absolute reset / return-to-origin" depending on the parameter (DI17 mode) setting. • When set at "ABS" (absolute reset); Turn ON to perform the absolute reset of the robot. The absolute reset is executed when this signal is switched from OFF to ON. The axis whose return-to-origin method is the mark method does not perform the absolute reset. Additionally, if the axis whose return-to-origin method is the mark method does not complete the return-to-origin, the absolute reset is not executed using the dedicated input. It is necessary that the robot controller is in the MANUAL mode. • When set at "ABS/ORG"(absolute reset/return-to-origin); Only the absolute type axis performs the absolute reset. The absolute reset is executed when this signal is switched from OFF to ON. When this signal is switched from OFF to ON, the incremental type axis performs the return-to-origin and the semi-absolute type axis performs the absolute search operation. When the absolute type axis, incremental type axis, and semi-absolute type axis are mixed, the incremental type axis and semi-absolute type axis perform the return-to-origin after the absolute type axis has performed the absolute reset.

Address	Signal name		Description
(n+34)0 ~ (n+34)7	SI(20) ~ SI(27)	General-purpose input	Refers to the SI port value, executes the WAIT command, and uses the ON/OFF state of the general-purpose input.
(n+35)0 ~ (n+35)7	SI(30) ~ SI(37)	General-purpose input	
~	~	~	
(n+44)0 ~ (n+44)7	SI(150) ~ SI(157)	General-purpose input	

n : Start address of the output area assigned to the master module

NOTE

- When the RCX141 or RCX221 is changed to the RCX240 and SI(17) needs to be used for the return-to-origin input, SI(17) is used for "absolute reset/return-to-origin".
- When the RCX240 is used with a robot whose axis configuration includes the absolute type, incremental type, and semi-absolute type axes and SI(17) is used for "absolute reset/return-to-origin", the absolute reset is performed for the absolute reset type axis each time the return-to-origin is performed for the incremental type or semi-absolute type axis.
  So, when the robot axis configuration includes the absolute type, incremental type, and semi-absolute type axes, it is recommended to perform the absolute reset with SI(17) and return-to-origin with SI(14).
- The return-to-origin input and absolute reset input can also be executed in the AUTO mode by changing the execution level. For details, refer to the user's manual for controller.

Word input

Address	Name			Description
n		SIW(0)	Dedicated input	Used as the remote command area.
n+2		SIW(1)		Used as the command data area of the remote command.
n+4	SID(2)	SIW(2)	General-purpose input	Used to input the word or double word data from the SIW or SID port. Or, used as the command data area of the remote command.
n+6		SIW(3)		
n+8	SID(4)	SIW(4)		
n+10		SIW(5)		
n+12	SID(6)	SIW(6)		
n+14		SIW(7)		
n+16	SID(8)	SIW(8)		
n+18		SIW(9)		
n+20	SID(10)	SIW(10)		
n+22		SIW(11)		
n+24	SID(12)	SIW(12)		
n+26		SIW(13)		
n+28	SID(14)	SIW(14)		
n+30		SIW(15)		

n : Start address of the output area assigned to the master module

Word output

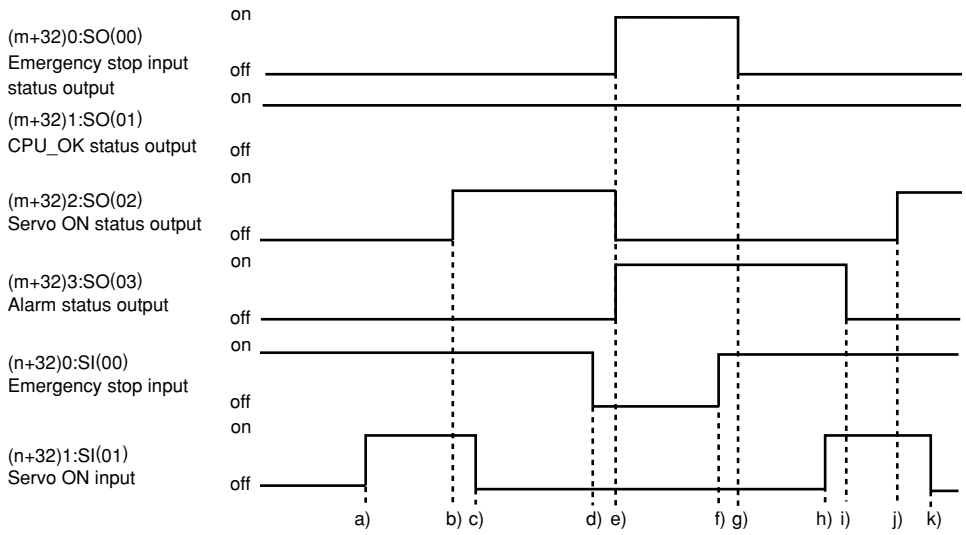
Address	Name			Description
m		SOW(0)	Dedicated output	Used as the status area of the remote command.
m+2		SOW(1)		Used as the error code area of the remote command.
m+4	SOD(2)	SOW(2)	General-purpose output	Used to output the word or double word data from the SOW or SOD port. Or, used as the response area of the remote command.
m+6		SOW(3)		
m+8	SOD(4)	SOW(4)		
m+10		SOW(5)		
m+12	SOD(6)	SOW(6)		
m+14		SOW(7)		
m+16	SOD(8)	SOW(8)		
m+18		SOW(9)		
m+20	SOD(10)	SOW(10)		
m+22		SOW(11)		
m+24	SOD(12)	SOW(12)		
m+26		SOW(13)		
m+28	SOD(14)	SOW(14)		
m+30		SOW(15)		

m : Start address of the input area assigned to the master module

3. Dedicated input/output signal timing chart

3.1 Servo ON and emergency stop

Servo ON and emergency stop



23501-S6-00

CAUTION

- **Provide an interval of 100ms or more when turning the dedicated input from the master module to the controller ON and OFF. If the interval is too short, the dedicated input may not be recognized. (This also applies to the interval for the same dedicated inputs or different dedicated inputs.)**
- **Use this also if there is a dedicated output in response to the dedicated input from the master module to the controller.**

■ Initial servo ON process after power ON

a) Servo ON input ON is input

b) If not in the emergency stop state, output servo ON status ON is output

c) After confirming that servo ON status output is ON, servo ON input OFF is input

■ Shift to emergency stop

d) Emergency stop input OFF is input

e) Emergency stop input status ON and alarm status output ON are output
Servo ON status output OFF is output

■ Servo ON process from emergency stop status

f) Emergency stop input ON is input

g) Emergency stop input status output OFF is output

h) Servo ON input ON is input

i) Alarm status output OFF is output

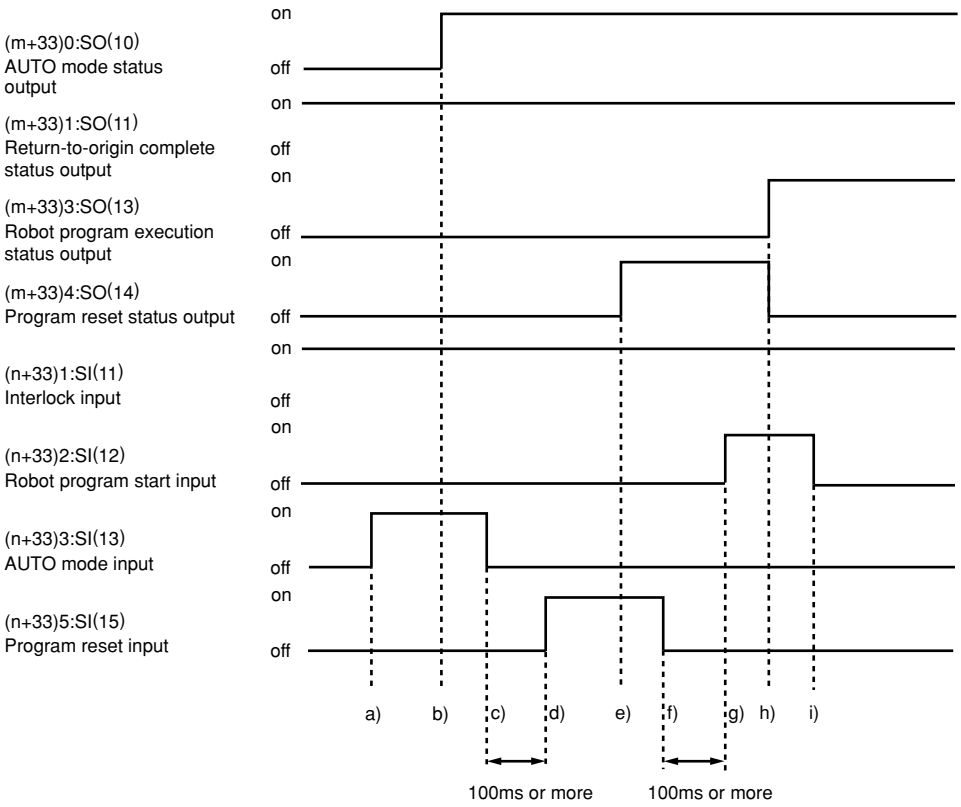
j) Servo ON status output ON is output

k) After confirming that servo ON status output is ON, servo ON input OFF is input

* The servo is OFF when the controller power is turned ON.

* When SAFE mode is enabled, dedicated inputs other than SI (00) and SI (11) might be disabled depending on service mode parameter setting unless service mode input signal is set to ON with SI (02) in the EtherNet/IP system.

AUTO mode changeover, program reset and program execution



23502-S6-00

CAUTION

- **Provide an interval of 100ms or more when turning the dedicated input from the master module to the controller ON and OFF. If the interval is too short, the dedicated input may not be recognized. (This also applies to the interval for the same dedicated inputs or different dedicated inputs.)**
- **Use this also if there is a dedicated output in response to the dedicated input from the master module to the controller.**

■ AUTO mode changeover process

a) AUTO mode input ON is input

b) AUTO mode status output ON is output

c) After confirming that the AUTO mode status output is ON, the AUTO mode input OFF is input

■ Program reset process

d) Program reset input ON is input

e) Program reset status output ON is output

f) After confirming that the program reset status output is ON, the program reset input OFF is input

■ Program execution process

g) Robot program start input ON is input

h) Program reset status output OFF is output
Robot program execution status output ON is output

i) After confirming that the robot program execution status output is ON, the robot program start input OFF is input

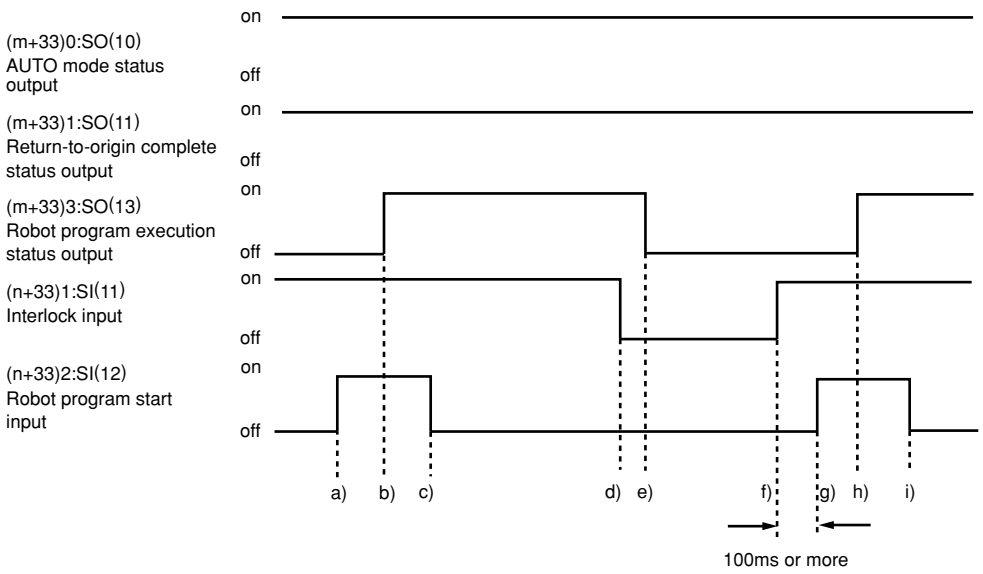
* The program cannot be executed if the emergency stop input and interlock input are OFF.

* If the return-to-origin complete status output is not ON, execution of the program may not be possible depending on the execution level setting value.

* When SAFE mode is enabled, dedicated inputs other than SI (00) and SI (11) might be disabled depending on service mode parameter setting unless service mode input signal is set to ON with SI (02) in the EtherNet/IP system.

3.3 Stopping with program interlock

Stopping with program interlock



23503-S6-00

CAUTION

- **Provide an interval of 100ms or more when turning the dedicated input from the master module to the controller ON and OFF. If the interval is too short, the dedicated input may not be recognized. (This also applies to the interval for the same dedicated inputs or different dedicated inputs.)**
- **Use this also if there is a dedicated output in response to the dedicated input from the master module to the controller.**

Program execution process

a) Robot program start input ON is input

b) Robot program execution status output ON is output

c) After confirming that the robot program execution status output is ON, the start input OFF is input

Program stop process using interlock input

d) Interlock input OFF is input

e) Robot program execution status output OFF is output

Program execution after stopping program with interlock input

f) Interlock input ON is input

g) Robot program start input ON is input

h) Robot program execution status output ON is output

i) After confirming that the robot program execution status output is ON, the start input OFF is input

* The program also stops when emergency stop input OFF is input. At this point, emergency stop input status ON and alarm status output ON are output, and servo ON status output OFF is output. To re-execute the program, servo ON process is required.

* When SAFE mode is enabled, dedicated inputs other than SI (00) and SI (11) might be disabled depending on service mode parameter setting unless service mode input signal is set to ON with SI (02) in the EtherNet/IP system.

4. EtherNet/IP compatible module specifications

Model / Spec. Item	EtherNet/IP compatible module		
Controller model	RCX240		
Software version	V10.50 or higher		
Network specifications	Conforms to Ethernet (IEEE 802.3).		
Applicable EtherNet/IP specifications	Volume 1 : Common Industrial protocol (CIP™) Edition 3.8 Volume 2 : EtherNet/IP Adaptation Edition 1.9		
Device type	Generic Device (Device No. 43)		
Data size	48 bytes each for input/output		
Transmission speed	10 Mbps/100 Mbps		
Connector specifications	RJ-45 connector (8-pole modular connector) 1 port		
Cable specifications	Refer to “4.1. LAN cable” in Chapter 2 of this guide.		
Max. cable length	100 m		
EtherNet/IP input/output points*1	Input (48 bytes in total)	byte 0-3 byte 4-31	Dedicated word input : 2 words General purpose word input : 14 words
		byte 32-33 byte 34-47	Dedicated bit input : 16 points General-purpose bit input : 96 points
	Output (48 bytes in total)	byte 0-3 byte 4-31	Dedicated word output : 2 words General-purpose word output : 14 words
		byte 32-33 byte 34-47	Dedicated bit output : 16 points General-purpose bit output : 96 points
Parallel external input	Regardless of the robot program, the master module and up to four ports can be controlled using the emulated serialization function.		
Settings, such as IP address	The settings are made with the programming box (RPB).		
Monitor LEDs	Activity, Network Status, Link, Module Status		
CPU BOARD ASSY	KX0-M4210-2XX (Refer to “3. Installing into the robot controller” in Chapter 1 of this guide.)		

*1: Controller’s I/O update intervals are 10 ms at shortest, but actual I/O update intervals may vary depending on the update time for the master station.

CAUTION

- **For the names and contents of the word and bit input/output signals, refer to the tables shown in the sections, "Profile" and "Details of input/output signals".**
- **The specifications and appearance are subject to change without prior notice due to continual improvement.**

5. EtherNet/IP specifications

■ General data

Applicable EtherNet/IP specifications	Volume 1 : Common Industrial protocol (CIP™) Edition 3.8 Volume 2 : EtherNet/IP Adaptation Edition 1.9
Vendor name	YAMAHA MOTOR CO.,LTD. (Vendor ID 636)
Device type	Generic Device (Device No. 43)
Product code	5
Product revision	1.1

Chapter 6 Appendix

Contents

1. Definitions of terms 6-1
2. EDS files 6-4

1. Definitions of terms

EtherNet/IP (Ethernet/Industrial Protocol)

This EtherNet/IP is a communication protocol that CIP (Common Industrial Protocol) is mounted on the Ethernet and TCP/IP.
The EtherNet/IP is jointly controlled by ODVA (Open DeviceNet Vendor Association) and CI (ContorolNet International).

CIP (Ethernet/Industrial Protocol)

This CIP is a protocol for the application layer that does not depend on the physical layer used for the EtherNet/IP or DeviceNet.
The CIP provides the standard object that can access to data and includes functions necessary for industrial network units.

TCP/IP (Transmission Control Protocol / Internet Protocol)

This TCP/IP is a standard protocol for the Internet communication. The TCP/IP is a generic name of multiple protocol groups that use the TCP and IP protocols as a core. All computers and personal computers that can access to the Internet use the TCP/IP protocol.

Ethernet

Simply put, this Ethernet is a kind of standard for the hardware related to the network system.
The Ethernet is a network that was invented by Xerox in the U.S.A. in the early 1970s. Presently, the Ethernet is national-standardized as IEE802.3. The specifications are classified into 10BASE-2, 10BASE-5, and 10BASE-T by the transmission cable type. The maximum cable length or the maximum number of connections may vary depending on the specifications. The EtherNet/IP compatible module for the RCX240 uses the 10BASE-T specifications.
Protocols generally used for the Ethernet are NetBEUI and IPX/SPX in addition to the TCP/IP protocol.
Additionally, the features of the Ethernet are that the CSMA/CD method is used for the data transmission method.

CSMA/CD (Carrier Sense Multiple Access with Collision Detection)

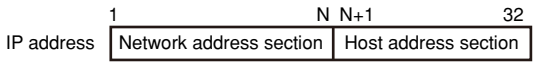
This CSMA/CD is a signal transmission method that the data transmission method called "CSMA" is combined with the transmission troubleshooting method called "CD".
Since the CSMA commonly uses multiple units connected to the network only with one transmission cable, it checks the network working status to confirm the transmission ready status before transmitting the data.
The CD waits for randomly determined time, and then transmits the data again if data collision occurs in the network.
Therefore, many units can be connected to the Ethernet that uses the CSMA/CD method, but the real time transmission is not guaranteed since the transmission waiting or re-transmission occurs.

IP address

The IP address is a unique number assigned to each device to identify that device on the network and prevent the same number from being used by different devices. (More accurately, an IP address is assigned to each network interface, since once device may sometimes be installed with multiple network interfaces.) In a TCP/IP protocol, the data transmit source and destination are specified by this IP address. The IP address consists of 32 bits (4 bytes) so can be expressed with this number without changes, however it is normally expressed as a decimal number separated by periods (.) at each byte (in other words, four sets of numbers separated by periods). An IP address of 0xC0A80002, for example, is normally expressed as 192.168.0.2.

The IP address is actually comprised of 2 address sections. One section is the network address. The network address is the address of the network itself. The other section is the host address section. The host address is an address for identifying each device on that network. The IP address, as shown below, uses the first through the Nth bits as the network address, and the N+1 bit through 32nd bit as the host address. (The value of N is determined by the subnet mask.)

IP address (1)

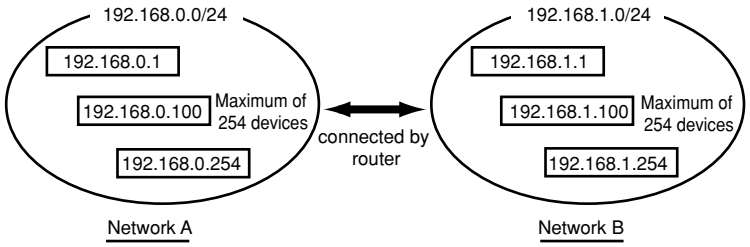


23601-S6-00

In an IP address of 192.168.0.2, for example, if the N value (network length) is 24 bits, then the network address section is 192.168.0, and the host address section is 2. Generally, in a network address, the host address section is 0 and the network length is listed behind the address. In the above example, this would be shown as 192.168.0.0/24.

One network can be connected with as many devices as there are addresses to identify them. However, host address bits having all zeroes (0), or all ones (1) are reserved and so cannot be used. In the above example, though the host address can identify 256 devices, the numbers 0 and 255 cannot be used so the maximum number of devices that can actually be connected is 254.

IP address (2)



23602-S6-00

Any company (organization) can freely select a host address but when attempting to connect their network to the Internet, that company (organization) cannot select the network address on their own. An application to acquire a network address must be made to the NIC (in Japan, JPNIC). If connecting one's network to the Internet is not necessary, then any company can freely select a network address, as well as a host address.

If there is no need to connect to the Internet, then use of the following addresses is allowed.

10.0.0.0 through 10.255.255.255	(1 unit of class A)
172.16.0.0 through 172.31.255.255	(16 units of class B)
192.168.0.0 through 192.168.255.255	(256 units of class C)

An address acquired by making application to NIC on the other hand is referred to as a global address.

Subnet mask

The subnet mask is used to separate the IP address into a network address section and a host address section. The network address bit is set to 1, and the host address bit is set to 0. The subnet mask, just like the IP address is expressed as a decimal number of 32 bits (4 bytes) with each byte separated by a period (or four sets of numbers separated by periods). So if the subnet mask is 255.255.255.0, then the network address section is 24 bits.

A company (organization) is generally assigned only one network address when applying to the NIC for an IP address. The company making the application falls within one of classes A, B or C depending on the scale of the company. Class B for example, has a network length of 16 bits and can be assigned a network allowing connection of up to 65533 devices. However, unless changes are made, this network cannot efficiently perform the required managing and processing tasks. So such a network is normally set with subnet masks to divide it into an appropriate number of smaller networks. When a class B network for example, is set with a subnet mask of 255.255.255.0, a total of 256 settings can be made allowing up to 254 devices to be connected.

HUB

A HUB is a device used for connecting devices such as PCs by way of a 10BASE-T network. The HUB has multiple ports that allow connecting modular jacks and twisted pair cables fitted with these modular jacks connect to the HUB from each device.
The HUB may have different type connectors depending on whether the HUB is for 10BASE-2 or 10BASE-5. Various types of networks can be constructed by means of these HUBs.

Router

The router is a device for mutually connecting networks together. The router is controlled based on a sophisticated process. The router sends data with an external destination from an internal network to an external network, and sends data received from an external network, to an internal network. Designated data is discarded in a filtering process to help maintain network safety.

The router IP address is set as the gateway address in each network device. This setting allows data to be correctly sent and received by each device on the network.

SAFE mode setting

When the SAFE mode setting is enabled, the SERVICE mode input is made valid so that the safety functions, such as operating speed limits in the MANUAL mode can be used. The SAFE mode setting is determined at the time of shipping. The SAFE mode setting is always enabled for controllers compatible with the CE marking.

SERVICE mode

This mode becomes valid when the SAFE mode setting is enabled. The SERVICE mode can be controlled by service mode input signals.

SAFETY connector

This SAFETY connector is a controller connector that connects the emergency stop input and service mode input.

STD.DIO connector

This STD.DIO connector is a controller connector that connects the dedicated input/output signals or general-purpose input/output signals.

Bit information

This bit information can be handled by the EtherNet/IP compatible module.

Word information

This word information can be handled by the EtherNet/IP compatible module.

Little endian

This little endian is a method that substitutes the LSB into the memory at low-order address and refers to the LSB when the word information data is handed as double word data.
For example, when the value "00012345h" is substituted into SOD (2), "2345h" is substituted into SOW (2) of the first word and "0001h2 into SOW (3) of the second word.

2. EDS files

The contents of an EDS file are shown below just for your reference.

```
$ EZ-EDS Version 3.9 Generated Electronic Data Sheet

[File]
        DescText = "YAMAHA RCX EtherNet/IP";
        CreateDate = 03-30-2011;
        CreateTime = 12:07:19;
        ModDate = 05-18-2011;
        ModTime = 14:00:47;
        Revision = 1.1;

[Device]
        VendCode = 636;
        VendName = "YAMAHA Motor Co., Ltd";
        ProdType = 43;
        ProdTypeStr = "Generic Device";
        ProdCode = 5;
        MajRev = 1;
        MinRev = 1;
        ProdName = "YAMAHA ROBOT RCX EIP";
        Catalog = "YAMAHA ROBOT RCX EIP";

[Device Classification]
        Class1 = EtherNetIP;

[Params]
        Param1 =
                0,              $ first field shall equal 0
                ,,              $ path size,path
                0x0000,             $ descriptor
                0xC7,           $ data type : 16-bit Unsigned Integer
                2,              $ data size in bytes
                "Output Size",      $ name
                "",             $ units
                "",             $ help string
                0,504,48,           $ min, max, default data values
                ,,,,            $ mult, dev, base, offset scaling not used
                ,,,,            $ mult, dev, base, offset link not used
                ;               $ decimal places not used

        Param2 =
                0,              $ first field shall equal 0
                ,,              $ path size,path
                0x0000,             $ descriptor
                0xC7,           $ data type : 16-bit Unsigned Integer
                2,              $ data size in bytes
                "Input Size",       $ name
                "",             $ units
                "",             $ help string
                0,504,48,           $ min, max, default data values
                ,,,,            $ mult, dev, base, offset scaling not used
                ,,,,            $ mult, dev, base, offset link not used
                ;               $ decimal places not used

        Param3 =
                0,              $ reserved, shall equal 0
                ,,              $ Link Path Size, Link Path
                0x0000,             $ Descriptor
```

```
            0xC8,               $ Data Type
            4,                  $ Data Size in bytes
            "RPI Range",          $ name
            "",                 $ units
            "",                 $ help string
            2000,3200000,150000,    $ min, max, default data values
            ,,,,                $ mult, div, base, offset scaling
            ,,,,                $ mult, div, base, offset links
            ;                   $ decimal places

[Assembly]
      Revision = 2;
      Assem100 =
            "OUTPUT",
            ,
            504,
            0x0000,
            ,,
            4032,;

      Assem150 =
            "INPUT",
            ,
            504,
            0x0000,
            ,,
            4032,;

[Connection Manager]
      Connection1 =
        0x04030002,         $ 0-15    = supported transport classes
                            $ 16      = trigger: cyclic
                            $ 17      = trigger: change of state
                            $ 18      = trigger: application
                            $ 19-23   = trigger: reserved
                            $ 24      = transport type: listen-only
                            $ 25      = transport type: input-only
                            $ 26      = transport type: exclusive-owner
                            $ 27      = transport type: redundant-owner
                            $ 28-30   = reserved
                            $ 31      = Client = 0 / Server = 1
        0x44640405,         $ 0       = O->T fixed size supported
                            $ 1       = O->T variable size supported
                            $ 2       = T->O fixed size supported
                            $ 3       = T->O variable size supported
                            $ 4-5     = O->T number of bytes per slot (obsolete)
                            $ 6-7     = T->O number of bytes per slot (obsolete)
                            $ 8-10    = O->T Real time transfer format
                            $ 11      = reserved
                            $ 12-14   = T->O Real time transfer format
                            $ 15      = reserved
                            $ 16      = O->T connection type: NULL
                            $ 17      = O->T connection type: MULTICAST
                            $ 18      = O->T connection type: POINT2POINT
                            $ 19      = O->T connection type: reserved
                            $ 20      = T->O connection type: NULL
                            $ 21      = T->O connection type: MULTICAST
                            $ 22      = T->O connection type: POINT2POINT
                            $ 23      = T->O connection type: reserved
                            $ 24      = O->T priority: LOW
                            $ 25      = O->T priority: HIGH
                            $ 26      = O->T priority: SCHEDULED
                            $ 27      = O->T priority: reserved
                            $ 28      = T->O priority: LOW
```

```
                    $ 29     = T->O priority: HIGH
                    $ 30     = T->O priority: SCHEDULED
                    $ 31     = T->O priority: reserved
    Param3,Param1,Assem100, $ O->T RPI, size, format
    Param3,Param2,Assem150, $ T->O RPI, size, format
    ,,              $ config #1 size, format
    ,,              $ config #2 size, format
    "Exclusive Owner",    $ Connection Name
    "",             $ help string
    "20 04 24 C5 2C 96 2C 64";   $ Path
Connection2 =
  0x02030002,       $ 0-15   = supported transport classes
                    $ 16     = trigger: cyclic
                    $ 17     = trigger: change of state
                    $ 18     = trigger: application
                    $ 19-23  = trigger: reserved
                    $ 24     = transport type: listen-only
                    $ 25     = transport type: input-only
                    $ 26     = transport type: exclusive-owner
                    $ 27     = transport type: redundant-owner
                    $ 28-30  = reserved
                    $ 31     = Client = 0 / Server = 1
  0x44640305,       $ 0      = O->T fixed size supported
                    $ 1      = O->T variable size supported
                    $ 2      = T->O fixed size supported
                    $ 3      = T->O variable size supported
                    $ 4-5    = O->T number of bytes per slot (obsolete)
                    $ 6-7    = T->O number of bytes per slot (obsolete)
                    $ 8-10   = O->T Real time transfer format
                    $ 11     = reserved
                    $ 12-14  = T->O Real time transfer format
                    $ 15     = reserved
                    $ 16     = O->T connection type: NULL
                    $ 17     = O->T connection type: MULTICAST
                    $ 18     = O->T connection type: POINT2POINT
                    $ 19     = O->T connection type: reserved
                    $ 20     = T->O connection type: NULL
                    $ 21     = T->O connection type: MULTICAST
                    $ 22     = T->O connection type: POINT2POINT
                    $ 23     = T->O connection type: reserved
                    $ 24     = O->T priority: LOW
                    $ 25     = O->T priority: HIGH
                    $ 26     = O->T priority: SCHEDULED
                    $ 27     = O->T priority: reserved
                    $ 28     = T->O priority: LOW
                    $ 29     = T->O priority: HIGH
                    $ 30     = T->O priority: SCHEDULED
                    $ 31     = T->O priority: reserved
    Param3,0,,          $ O->T RPI, size, format
    Param3,Param2,Assem100, $ T->O RPI, size, format
    ,,              $ config #1 size, format
    ,,              $ config #2 size, format
    "Input Only",       $ Connection Name
    "",             $ help string
    "20 04 24 C5 2C C6 2C 64";   $ Path
Connection3 =
  0x01030002,       $ 0-15   = supported transport classes
                    $ 16     = trigger: cyclic
                    $ 17     = trigger: change of state
                    $ 18     = trigger: application
                    $ 19-23  = trigger: reserved
                    $ 24     = transport type: listen-only
                    $ 25     = transport type: input-only
                    $ 26     = transport type: exclusive-owner
```

```
                        $ 27      = transport type: redundant-owner
                        $ 28-30   = reserved
                        $ 31      = Client = 0 / Server = 1
        0x44240305,     $ 0       = O->T fixed size supported
                        $ 1       = O->T variable size supported
                        $ 2       = T->O fixed size supported
                        $ 3       = T->O variable size supported
                        $ 4-5     = O->T number of bytes per slot (obsolete)
                        $ 6-7     = T->O number of bytes per slot (obsolete)
                        $ 8-10    = O->T Real time transfer format
                        $ 11      = reserved
                        $ 12-14   = T->O Real time transfer format
                        $ 15      = reserved
                        $ 16      = O->T connection type: NULL
                        $ 17      = O->T connection type: MULTICAST
                        $ 18      = O->T connection type: POINT2POINT
                        $ 19      = O->T connection type: reserved
                        $ 20      = T->O connection type: NULL
                        $ 21      = T->O connection type: MULTICAST
                        $ 22      = T->O connection type: POINT2POINT
                        $ 23      = T->O connection type: reserved
                        $ 24      = O->T priority: LOW
                        $ 25      = O->T priority: HIGH
                        $ 26      = O->T priority: SCHEDULED
                        $ 27      = O->T priority: reserved
                        $ 28      = T->O priority: LOW
                        $ 29      = T->O priority: HIGH
                        $ 30      = T->O priority: SCHEDULED
                        $ 31      = T->O priority: reserved
            Param3,0,,            $ O->T RPI, size, format
            Param3,Param2,Assem100, $ T->O RPI, size, format
            ,,                    $ config #1 size, format
            ,,                    $ config #2 size, format
            "Listen Only",        $ Connection Name
            "",                   $ help string
            "20 04 24 C5 2C C7 2C 64";    $ Path

[Port]
        Port1 =
            TCP,                  $ port type name
            "TCP/IP",             $ name of port
            "20 F5 24 01",        $ instance one of the TCP/IP interface object
            2;                    $ port number

[Capacity]
        MaxCIPConnections = 63;       $ Max CIP cnx - all classes
        TSpec1 = TxRx, 1, 2000;       $ Packets per sec @ 1 bytes
        TSpec2 = TxRx, 504, 2000;     $ Packets per sec @ 504 bytes
```

Remote command guide

Contents

1. Remote command format A-1

1.1 Remote command specifications A-1

1.2 Remote status A-2

2. Sending and receiving remote commands A-3

3. Remote command & remote status tables A-4

4. Remote command description A-8

4.1 Special commands A-8

4.1.1 Status reset command A-8
4.1.2 Main robot current position reference command A-8
4.1.3 Sub robot current position reference command A-9

4.2 Category 1 remote commands A-11

4.2.1 MOVE command A-12
4.2.1.1 PTP designation A-12
4.2.1.2 Arch designation A-15
4.2.1.3 Linear interpolation A-18
4.2.1.4 Circular interpolation A-21
4.2.1.5 Direct PTP designation (millimeter units) A-24
4.2.1.6 Direct PTP designation (pulse units) A-27
4.2.2 MOVEI command A-30
4.2.2.1 PTP designation A-30
4.2.2.2 Direct PTP designation (millimeter units) A-33
4.2.2.3 Direct PTP designation (pulse units) A-36
4.2.3 DRIVE command A-39
4.2.3.1 Point designation A-39
4.2.3.2 Direct designation (millimeter units) A-42
4.2.3.3 Direct designation (pulse units) A-45
4.2.4 DRIVEI command A-48
4.2.4.1 Point designation A-48
4.2.4.2 Direct designation (millimeter units) A-51
4.2.4.3 Direct designation (pulse units) A-54
4.2.5 Pallet movement command A-57
4.2.5.1 PTP designation A-57
4.2.5.2 Arch designation A-60
4.2.6 Jog movement command A-63
4.2.7 Inching movement command A-66
4.2.8 Point teaching command A-69
4.2.9 Absolute reset movement command A-70
4.2.10 Absolute reset command A-71
4.2.11 Return-to-origin command A-72
4.2.12 Servo command A-74
4.2.13 Manual speed change command A-75
4.2.14 Auto speed change command A-76

4.2.15 Program speed change command A-77
4.2.16 Shift designation change command A-78
4.2.17 Hand designation change command A-79
4.2.18 Arm designation change command A-80
4.2.19 Point display unit designation command A-81
4.3 Category 2 remote commands A-83
4.3.1 Point-related command A-84
4.3.1.1 Point data definition A-84
4.3.1.2 Point data reference A-86
4.3.2 Point comment-related command A-88
4.3.2.1 Point comment data definition A-88
4.3.2.2 Point comment data reference A-90
4.3.3 Pallet-related command A-92
4.3.3.1 Pallet data definition A-92
4.3.3.2 Pallet data reference A-93
4.3.4 Shift-related command A-95
4.3.4.1 Shift data definition A-95
4.3.4.2 Shift data reference A-97
4.3.5 Hand-related command A-99
4.3.5.1 Hand data definition A-99
4.3.5.2 Hand data reference A-101
4.4 Category 3 remote commands A-103
4.4.1 Static variable-related command A-104
4.4.1.1 Assigning a numerical value to a static variable A-104
4.4.1.2 Assigning a variable to a static variable A-106
4.4.1.3 Arithmetic operation using numerical data on static variable A-108
4.4.1.4 Arithmetic operation using variable on static variable A-110
4.4.1.5 Static variable value reference A-112
4.4.2 Parameter-related command A-114
4.4.2.1 Assigning a value to a parameter A-114
4.4.2.2 Parameter value reference A-116
4.4.3 Point-related command A-118
4.4.3.1 Assigning a point to a parameter A-118
4.4.3.2 Point addition/subtraction A-120
4.4.3.3Assigning a pallet point A-122
4.4.4 Element assignment command A-124
4.4.4.1 Assigning to a point element A-124
4.4.4.2 Assigning to a shift element A-126
4.5 Category 4 remote commands A-128
4.5.1 I/O port commands A-129
4.5.1.1 Assigning a numerical value to an I/O port A-129
4.5.1.2 I/O port reference A-131
4.6 Category 5 remote commands A-133
4.6.1 Execution program designation A-134
4.6.2 Program execution A-135
4.6.3 Program reset A-136
4.6.4 Program task switching A-137
4.6.5 Program execution information reference A-138
4.7 Category 6 remote commands A-140
4.7.1 Version information reference A-141

4.7.2 Controller configuration reference A-142
4.7.3 Servo status reference A-144
4.7.4 Absolute reset status reference A-145
4.7.5 Current position reference A-147
4.7.5.1 Pulse designation A-147
4.7.5.2 Millimeter designation A-148
4.7.6 Task status reference A-150
4.7.7 Task execution line reference A-151
4.7.8 Message reference A-152
4.7.9 Speed status reference A-153
4.7.10 Arm designation status reference A-154
4.7.11 Arm status reference A-155
4.7.12 Service mode status reference A-156
4.7.13 Point unit status reference A-157
4.7.14 Return-to-origin status reference A-158

1. Remote command format

Using the EtherNet/IP compatible module allows issuing commands directly from the PLC (programmable logic controller). Commands such as MOVE commands can now be run that were impossible to execute up until now without using the robot program or RS-232C port.

CAUTION

- **Remote commands are compatible with the RCX240 software version 10.52 or higher.**
- **To use remote commands, the "Remote_cmd SI05" parameter in SYSTEM > PARAM > OP.BRD mode must be set to "VALID" in advance. For details, refer to "3.2. Setting the "Remote_cmd SI05" function" in Chapter 2 of the EtherNet/IP compatible module guide and the user's manual for controller.**

1.1 Remote command specifications

Functions such as shown below are assigned to each remote register.

Output (remote → master)			Input (remote ← master)	
Address	Contents		Address	Contents
m	Status		n	Execute command code
	Normal end	Abnormal end		
m+2	Response	Error code	n+2	Command data
m+4		Additional information	n+4	
m+6			n+6	
to			to	
m+30			n+30	

m : Start address of the input area assigned to the master module
n : Start address of the output area assigned to the master module

NOTE

Remote commands must be held until the status changes to a normal end (0x0200) or an abnormal end (0x4000). If a remote command is changed before the status changes to an end, the status of the remote command executed will not be reflected.

- Remote commands are run by assigning the command codes to the "n", and command data to the n+2 to n+30. When the controller receives the remote command, it starts the processing and sends the status (results) and its other information to the PLC by way of the "m" and m+2 to m+30. When the remote command ends, assign the status reset command (0x0000 (hexadecimal)) to the "n" to clear the status. The remote command can be run when in command ready status (0x0000 (hexadecimal)).
- Command data to be added to remote commands differs according to the particular remote command. For details, Refer to "4. Remote command description" in this guide. Command data must always be entered before trying to set the remote command.
- Contents of the remote command response sent as the remote command results differ according to the particular remote command. For details, Refer to "4. Remote command description" in this guide.
- Data is set in binary code. When setting two pieces of 8-bit data such as character code data, set the upper bit data into the higher address. If the data size is greater than 16 bits, set the upper bit data into the higher address. (little endian)
  For example, to set "12" in n+8, enter 0x3231 (hexadecimal)
  (character code: "1" = 0x31, "2" = 0x32)
  For example, to set 0x01234567 (hexadecimal) (=19,088,743) in the n+8 and n+10 registers, set 0x0123 (hexadecimal) in n+10 and set 0x4567 (hexadecimal) in n+8.
- The status code is sent to "m" when the remote command ends correctly.
- When the remote command ends incorrectly, an error code is sent to m+2 and additional information is sent to m+4 as a response. The error group number is displayed in the upper 8 bits of the error code and the error category number is displayed in the lower 8 bits. The additional information section appears in the upper 8 bits of additional information and a detail value for the additional information appears in the lower 8 bits. See the troubleshooting section of the robot controller user's manual for description of the error group number and error category number.
  For example, when 0x0201 (hexadecimal) was set in m+2, this shows that a "soft limit over" error has occurred. When 0x0001 (hexadecimal) is set in m+4, it indicates that Axis 1 of the controller is selected.

1.2 Remote status

The controller starts processing when the remote command is received and sends the status (results) to the PLC by way of "m".

■ Remote status list

Status contents				Meaning
m	m+2	m+4	From m+6	
0x0000	0x0000			Command ready status
0x0100	0x0000			Command run status
0x0200	Response data			Normal end status
0x4000	Error code	Additional information	0x0000	Abnormal end status

m : Start address of the input area assigned to the master module

NOTE

Remote commands must be held until the status changes to a normal end (0x0200) or an abnormal end (0x4000). If a remote command is changed before the status changes to an end, the status of the remote command executed will not be reflected.

■ Code 0x0000 .....Command ready status

Indicates a state where remote command is not being run and a new remote command can be received. Remote status must always be set to command ready status (0x0000) in order to execute a remote command. To change the remote status to command ready status (0x0000), run the status reset command (0x0000).

■ Code 0x0100 ..... Command run status

Indicates a state where the controller has received a remote command and is in command run status.
In some cases the command run status (0x0100) might not be sent to the PLC due to problems caused by a short remote command execution time versus the controller scan time (10 ms).

■ Code 0x0200 ..... Normal end status

Indicates a state where the remote command was run correctly.
Category 5 (key operation command) indicates command was received as a key operation command. The actual key operation sometimes might be in progress.

■ Code 0x4000 .....Abnormal end status

Indicates remote command ended abnormally.
Error number and error additional information on the error that occurred are sent to m+2 and m+4.

- **Error code m+2**

  Shows the error code for error causing command to end abnormally.
  Upper 8 bits show the group number and lower 8 bits show the category number.

- **Additional information m+4**

  Shows additional information if present in error code, such as axis number causing error.
  Upper 8 bits show the section number of additional information and lower 8 bits show a detail value.

Section No.	Contents
00	Actual axis d
01	Axis-d of main robot
02	Axis-d of sub robot
04	Main robot
05	Sub robot
09	d task

Here, d shows a detail value for the additional information.

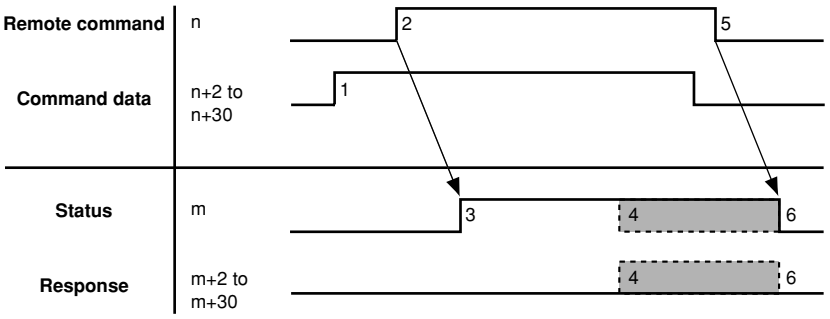
* For example, 0x0C02 is set in m+2 as the error code when the remote command was interrupted by an interlock signal.

* For information on the error code, refer to the error message section of the robot controller user's manual.

2. Sending and receiving remote commands

Remote register transmit and receive is performed as follows.

Sending and receiving remote commands



m : Start address of the input area assigned to the master module
n : Start address of the output area assigned to the master module
("m" indicates the data direction from remote to master, and "n" the data direction from master to remote.)

93201-S6-00

NOTE

Remote commands must be held until the status changes to a normal end (0x0200) or an abnormal end (0x4000). If a remote command is changed before the status changes to an end, the status of the remote command executed will not be reflected.

1. Command data setting
2. Remote command setting
3. Status shifts to command run status (0x0100).
   (If the command is quickly executed, status may sometimes shift to normal end status (0x0200) without changing to command run status (0x0100).)
4. Shifts to response change and normal end status (0x0200) or to abnormal end status (0x4000).
5. Status reset command (0x0000) setting
6. Status and response shifts to command ready status.

Example: Typical transmit/receive when running a PTP movement command (all axes, program speed 50%) to point 19 is shown below.

1. To run the PTP movement command for the designated point, enter the value in the registers shown below.
   n+2 : command flag (0x0004 = speed setting)
   n+6 : speed setting (0x0032=50%)
   n+8 : point setting (0x0013= point 19)
2. Enter the PTP movement command (0x0001) for the designated point into the "n".
3. The robot controller receives the remote command and starts running it if the command code and command data can be executed. Status now shifts to command run status (0x0100). The robot moves to the position designated as point 19 at the program speed (50% of normal speed). If the command cannot be executed, status shifts to abnormal end status (0x4000) and m+2 changes to an error code.
4. When finished executing the remote command, status changes to normal end status (0x0200). Response information is changed at the same time if present.
5. The current remote command has now finished, so set the status reset command (0x0000) in "n" in order to issue the next command.
6. The status and response shift to command ready status (0x0000).

3. Remote command & remote status tables

Remote commands and remote status codes are shown in hexadecimal notation.

■ Remote Command

Command contents		Meaning
Category	n	
Special	0x0000	Status reset command
	0x8000	Main robot current position reference command
	0x4000	Sub robot current position reference command
1	0x00nn	Movement command and associated command (including commands supported with I/O command)
2	0x01nn	Definition and reference command
3	0x02nn	Arithmetic command
4	0x03nn	I/O port command
5	0x04nn	Key operation command
6	0x05nn	Data handling command

n : Start address of the output area assigned to the master module
("n" indicates the data direction from master to remote.)

* nn is determined by the particular remote command.

■ Remote Status

Status contents				Meaning
m	m+2	m+4	From m+6	
0x0000	0x0000			Command ready status
0x0100	0x0000 or response data			Command run status
0x0200	Response data			Normal end status
0x4000	Error code	Additional information	0x0000	Abnormal end status

m : Start address of the input area assigned to the master module
("m" indicates the data direction from remote to master.)

Category 1

No.	Command contents			Command code n	
				Main robot	Sub robot
1-1	MOVE command	PTP point designation		0x0001	0x0081
		Arch designation		0x0002	0x0082
		Linear interpolation		0x0003	0x0083
		Circular interpolation		0x0004	0x0084
		Direct PTP designation	Millimeter units	0x0006	0x0086
			Pulse units	0x0007	0x0087
1-2	MOVEI command	PTP point designation		0x0009	0x0089
		Direct PTP designation	Millimeter units	0x000E	0x008E
			Pulse units	0x000F	0x008F
1-3	DRIVE command	Point designation		0x0010	0x0090
		Direct designation	Millimeter units	0x0012	0x0092
			Pulse units	0x0013	0x0093
1-4	DRIVEI command	Point designation		0x0014	0x0094
		Direct designation	Millimeter units	0x0016	0x0096
			Pulse units	0x0017	0x0097
1-5	Pallet command	PTP designation		0x0018	0x0098
		Arch designation		0x0019	0x0099
1-6	Jog movement command			0x0020	0x00A0
1-7	Inching movement command			0x0024	0x00A4
1-8	Point teaching command			0x0028	0x00A8
1-9	Absolute reset movement command			0x0030	0x00B0
1-10	Absolute reset command			0x0031	0x00B1
1-11	Return-to-origin command			0x0032	0x00B2
1-12	Servo command	On designation		0x0034	0x00B4
		Off designation		0x0035	0x00B5
		Free designation		0x0036	0x00B6
		Power-on designation		0x0037	
1-13	Manual speed change command			0x0038	0x00B8
1-14	Automatic speed change command			0x0039	0x00B9
1-15	Program speed change command			0x003A	0x00BA
1-16	Shift designation change command			0x003B	0x00BB
1-17	Hand designation change command			0x003C	0x00BC
1-18	Arm designation change command			0x003D	0x00BD
1-19	Point display unit designation command			0x003E	

* The DRIVE command (1-3) and DRIVEI command (1-4) are only valid for a single axis.
* The movement methods on the jog movement command (1-6) and inching movement command (1-7) will differ according to the point units that were specified.
* Point units for the point teaching command (1-8) will differ according to the point units that were specified.
* If no axis is specified, the absolute reset command (1-10) is executed on all axes (main robot + sub robot) in either case of command code 0x0031 or 0x00B1.
* If no axis is specified, the return-to-origin command (1-11) is executed on all axes (main robot + sub robot) in either case of command code 0x0032 or 0x00B2.
* The point unit designation command (1-19) is for use on the controller.

Category 2

No.	Command contents			Command code n
2-1	Point-related command	Point data definition		0x0100
		Point data reference		0x0101
2-2	Point comment-related command	Point comment data definition		0x0104
		Point comment data reference		0x0105
2-3	Pallet-related command	Pallet data definition		0x0108
		Pallet data reference		0x0109
2-4	Shift-related command	Shift data definition		0x010C
		Shift data reference		0x010D
2-5	Hand-related command	Hand data definition	Main robot	0x0110
			Sub robot	0x0190
		Hand data reference		0x0111

Category 3

No.	Command contents			Command code n
3-1	Static variable-related commands	Assignment	Value	0x0200
			Variable	0x0201
		Addition	Value	0x0204
			Variable	0x0205
		Subtraction	Value	0x0208
			Variable	0x0209
		Multiplication	Value	0x020C
			Variable	0x020D
		Division	Value	0x0210
			Variable	0x0211
		Reference	Variable	0x0214
3-2	Parameter-related command	Assignment	Main robot	0x0220
			Sub robot	0x02A0
		Reference	Main robot	0x0224
			Sub robot	0x02A4
3-3	Point-related command	Point assignment		0x0230
		Addition		0x0234
		Subtraction		0x0235
		Pallet point assignment		0x0238
3-4	Element assignment command	Point element assignmen	"x1" input format	0x0240
			"x100" input format	0x0241
		Shift element assignment	"x100" input format	0x0245

Category 4

No.	Command contents			Command code n
4-1	I/O port-related commands	Assignment	Port units	0x0300
			Bit units	0x0301
		Reference	Port units	0x0304

Category 5

No.	Command contents		Command code n
5-1	Execution program designation		0x0401
5-2	Program execution	Program execution	0x0402
		Program step execution	0x0403
		Program skip execution	0x0404
		Program next execution	0x0405
5-3	Program reset		0x0406
5-4	Program task switching		0x0407
5-5	Program execution information reference		0x0408

Category 6

No.	Command contents			Command code n
6-1	Version information reference			0x0501
6-2	Controller configuration reference			0x0502
6-3	Servo status reference			0x0503
6-4	Absolute reset status reference			0x0504
6-5	Current position reference	Pulse units	Main robot	0x0505
			Sub robot	0x0585
		Millimeter units	Main robot	0x0506
			Sub robot	0x0586
6-6	Task status reference			0x0507
6-7	Task execution reference			0x0508
6-8	Message reference			0x0509
6-9	Speed status reference			0x050A
6-10	Arm designation status reference			0x050B
6-11	Arm status reference			0x050C
6-12	Service mode status reference			0x050D
6-13	Point unit status reference			0x050E
6-14	Return-to-origin status reference			0x050F

4. Remote command description

4.1 Special commands

Special commands are used in applications different from other remote commands.
A list of those commands is shown below.

No.	Command contents	Command code n
1	Status reset command	0x0000
2	Main robot current position reference command	0x8000
3	Sub robot current position reference command	0x4000

4.1.1 Status reset command

This command is executed to set the status to command ready status (0x0000).
Remote commands cannot be executed unless in command ready status (0x0000). Therefore after executing a remote command, this command must always be executed before running the next command.

■ **Command**

Channel	Contents	Value
n	Command code	0x0000
n+2	Not used	0x0000
to		
n+30		

■ **Status**

Channel	Contents	Value
m	Status code	0x0000
m+2	Response	
to		
m+30		

4.1.2 Main robot current position reference command

Execute this command to obtain the main robot current position data.
This command returns normal end status (0x0200) as the status code, but continually executes the main robot current position reference command until the status reset command is run. Units for the current position obtained at this time are the same as the point unit system of the controller. This command is useful when constantly monitoring the main robot current position.

■ **Command**

Channel	Contents	Value
n	Command code	0x8000
n+2	Not used	0x0000
to		
n+30		

Status

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for the current position data obtained. Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the data in 32 bits. (little endian)
Data is shown in integers when units are in pulses.
Data is shown in integers (x100) when units are in millimeters.

Example:

Values are expressed as shown at right when controller display units are in millimeters and with:

Axis 1 = 200.01

Axis 3 = –123.45

Other axes = 0.00

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0001
m+8	0x4E21
m+10	0x0000
m+12	0x0000
m+14	0x0000
m+16	0xCFC7
m+18	0xFFFF
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.1.3 Sub robot current position reference command

Execute this command to obtain the sub robot current position data.
This command returns normal end status (0x0200) as the status code, but continually executes the sub robot current position reference command until the status reset command is run. Units for the current position obtained at this time are the same as the point unit system of the controller. This command is valid when constantly monitoring the sub robot current position.

Command

Channel	Contents	Value
n	Command code	0x4000
n+2	Not used	0x0000
to		
n+30		

Status

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for the current position data obtained. Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the data in 32 bits. (little endian)
Data is shown in integers when units are in pulses.
Data is shown in integers (x100) when units are in millimeters.

Example:

Values are expressed as shown at right when controller display units are in pulses and with:

Axis 1 = 123456

Axis 2 = -123

Other axes = 0

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0xE240
m+10	0x0001
m+12	0xFF85
m+14	0xFFFF
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2 Category 1 remote commands

These are remote commands mainly for movement commands. A list of these commands is shown below.

No.	Command contents			Command code n	
				Main robot	Sub robot
1	MOVE command	PTP point designation		0x0001	0x0081
		Arch designation		0x0002	0x0082
		Linear interpolation		0x0003	0x0083
		Circular interpolation		0x0004	0x0084
		Direct PTP designation	Millimeter units	0x0006	0x0086
			Pulse units	0x0007	0x0087
2	MOVEI command	PTP point designation		0x0009	0x0089
		Direct PTP designation	Millimeter units	0x000E	0x008E
			Pulse units	0x000F	0x008F
3	DRIVE command	Point designation		0x0010	0x0090
		Direct designation	Millimeter units	0x0012	0x0092
			Pulse units	0x0013	0x0093
4	DRIVEI command	Point designation		0x0014	0x0094
		Direct designation	Millimeter units	0x0016	0x0096
			Pulse units	0x0017	0x0097
5	Pallet command	PTP designation		0x0018	0x0098
		Arch designation		0x0019	0x0099
6	Jog movement command			0x0020	0x00A0
7	Inching movement command			0x0024	0x00A4
8	Point teaching command			0x0028	0x00A8
9	Absolute reset movement command			0x0030	0x00B0
10	Absolute reset command			0x0031	0x00B1
11	Return-to-origin command			0x0032	0x00B2
12	Servo command	On designation		0x0034	0x00B4
		Off designation		0x0035	0x00B5
		Free designation		0x0036	0x00B6
		Power-on designation		0x0037	
13	Manual speed change command			0x0038	0x00B8
14	Automatic speed change command			0x0039	0x00B9
15	Program speed change command			0x003A	0x00BA
16	Shift designation change command			0x003B	0x00BB
17	Hand designation change command			0x003C	0x00BC
18	Arm designation change command			0x003D	0x00BD
19	Point display unit designation command			0x003E	

m: Start address of the input area assigned to the master module
n : Start address of the output area assigned to the master module
("m" indicates the data direction from remote to master, and "n" the data direction from master to remote.)

* The DRIVE command (3) and DRIVEI command (4) are only valid for a single axis.
* The movement methods on the jog movement command (6) and inching movement command (7) will differ according to the point units that were specified.
* Point units for the point teaching command (8) will differ according to the point units that were specified.
* If no axis is specified, the absolute reset command (10) is executed on all axes (main robot + sub robot) in either case of command code 0x0031 or 0x00B1.
* If no axis is specified, the return-to-origin command (11) is executed on all axes (main robot + sub robot) in either case of command code 0x0032 or 0x00B2.
* The point unit designation command (19) is for use on the controller.

4.2.1 MOVE command

Execute this command group to move the robot to an absolute position.

4.2.1.1 PTP designation

This command moves the robot to a target position in PTP motion by specifying the point number.

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0001
			For sub robot	0x0081
n+2	Command flag	bit 0	Axis designation flag	a
		bit 2–bit 1	Speed designation flag	bb
		bit 14–bit 3	(0:Fixed)	0
		bit 15	Current position output designation flag	n
n+4	Specified axis to move	bit 0	Axis 1	0x00tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 15–bit 6	(0:Fixed)	
n+6	Specified speed			0xssss
n+8	Point numbe			0xpppp
n+10	Not used			0x0000
to				
n+30				

a :Specify in 1 bit whether all axes are designated.

Value	Meaning
0	All axes are specified.
1	One or more axes are specified.

bb :Specify the speed setting method in 2 bits.

Value	Meaning
00	Speed is not specified.
10	Speed is set in %.

n :Specify in 1 bit whether to output current position.

Value	Meaning
0	No output.
1	Output.

tt :Specify the axis to move in bit pattern using lower 8 bits. Valid when axis designation flag is 1.

ssss :Specify the movement speed in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

pppp :Specify the point number in 16 bits.
Specified range: 0 (=0x0000) to 9999 (=0x270F)

Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data. Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Specify the MOVE command with PTP designation as shown at right, when moving all axes of the main robot to point number 100 at 50% speed. The current position output is specified at this time.

Channel	Value
n	0x0001
n+2	0x8004
n+4	0x0000
n+6	0x0032
n+8	0x0064
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when controller display units are in pulses and with:

Axis 1 = 123456

Axis 2 = -123

Other axes = 0

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0xE240
m+10	0x0001
m+12	0xFF85
m+14	0xFFFF
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2.1.2 Arch designation

This command moves the robot to a target position in arch motion by specifying the point number, arch axis and arch data.

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0002
			For sub robot	0x0082
n+2	Command flag	bit 0	Axis designation flag	a
		bit 2–bit 1	Speed designation flag	bb
		bit 3	(0:Fixed)	0
		bit 4	Arch data unit flag	d
		bit 14–bit 5	(0:Fixed)	0
		bit 15	Current position output designation flag	n
n+4	Specified axis to move	bit 0	Axis 1	0xuutt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 7–bit 6	(0:Fixed)	
	Arch designation axis	bit 8	Axis 1	
		bit 9	Axis 2	
		bit 10	Axis 3	
		bit 11	Axis 4	
		bit 12	Axis 5	
		bit 13	Axis 6	
		bit 15–bit 14	(0:Fixed)	
n+6	Specified speed			0xssss
n+8	Point number			0xpppp
n+10	Not used			0x0000
n+12				
n+14				
n+16	Arch position data			0xqqqqqqqq
n+18				
n+20	Not used			0x0000
to				
n+30				

a :Specify in 1 bit how to designate axis.

Value	Meaning
0	All axes are specified.
1	One or more axes are specified.

bb :Specify the speed setting method in 2 bits.

Value	Meaning
00	Speed is not specified.
10	Speed is set in %.

d :Specify the arch data units in 1 bit.

Value	Meaning
0	Pulse units
1	Millimeter units

n :Specify in 1 bit whether to output current position.

Value	Meaning
0	No output.
1	Output.

tt :Specify the axis to move in bit pattern using lower 8 bits.
Valid when axis designation flag is 1.

uu :Specify the arch motion axis in bit pattern using upper 8 bits.
Specified arch axis is one axis only.

ssss :Specify the speed in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

pppp :Specify the point number in 16 bits.
Specified range: 0 (=0x0000) to 9999 (=0x270F)

qqqqqqqq :Specify the arch position in 32 bits. (little endian)
Data should be integers when units are in pulses.
Data should be integers (x100) when units are in millimeters.

Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data. Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Spmecify the MOVE command with arch designation as shown at right, when moving all axes of the main robot to point number 100 at 50% speed by way of a Z-axis arch position of 10.00mm. The current position output is specified at this time.

Channel	Value
n	0x0002
n+2	0x8014
n+4	0x0400
n+6	0x0032
n+8	0x0064
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x03E8
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when controller display units are in millimeters and with:

Axis 1 = 123.45
Axis 2 = -1.23
Axis 3 = 50.00
Axis 4 = 90.23
Other axes = 0.00

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0001
m+8	0x3039
m+10	0x0000
m+12	0xFF85
m+14	0xFFFF
m+16	0x1388
m+18	0x0000
m+20	0x233F
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2.1.3 Linear interpolation

This command moves the robot to a target position by linear interpolation by specifying the point number.

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0003
			For sub robot	0x0083
n+2	Command flag	bit 0	(0:Fixed)	0
		bit 2–bit 1	Speed designation flag	bb
		bit 4–bit 3	(0:Fixed)	0
		bit 5	Acceleration designation flag	d
		bit 6	Deceleration designation flag	e
		bit 14–bit 7	(0:Fixed)	0
		bit 15	Current position output designation flag	n
n+4	Not used			0x0000
n+6	Specified speed			0xssss
n+8	Point number			0xpppp
n+10 to n+18	Not used			0x0000
n+20	Acceleration designation			0xrrrr
n+22	Deceleration designation			0xrrrr
n+24 to n+30	Not used			0x0000

bb :Specify the speed setting method in 2 bits.

Value	Meaning
00	Speed is not specified.
10	Speed is set in %.
11	Speed is specified in mm/s.

d :Specify in 1 bit whether to set acceleration.

Value	Meaning
0	Acceleration is not specified.
1	Acceleration is specified.

e :Specify in 1 bit whether to set deceleration.

Value	Meaning
0	Deceleration is not specified.
1	Deceleration is specified.

n :Specify in 1 bit whether to output current position.

Value	Meaning
0	No output.
1	Output.

ssss :Specify the speed in 16 bits.
Specified range: Speed % : 1 (=0x0001) to 100 (=0x0064)
Specified speed in mm/s : 1 (=0x0001) to 1000 (=0x03E8)

pppp :Specify the point number in 16 bits.
Specified range: 0 (=0x0000) to 9999 (=0x270F)

rrrr :Specify the acceleration and deceleration in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data. Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Specify the MOVE command with linear interpolation as shown at right, when moving all axes of the main robot to point number 100 at a speed of 200 mm/s and at 50% acceleration. The current position output is specified at this time.

Channel	Value
n	0x0003
n+2	0x8026
n+4	0x0000
n+6	0x00C8
n+8	0x0064
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0032
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when controller display units are in millimeters and with:

Axis 1 = 123.45

Axis 2 = -1.23

Axis 3 = 50.00

Axis 4 = 90.23

Other axes = 0.00

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0001
m+8	0x3039
m+10	0x0000
m+12	0xFF85
m+14	0xFFFF
m+16	0x1388
m+18	0x0000
m+20	0x233F
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2.1.4 Circular interpolation

This command moves the robot to a target position by circular interpolation by specifying two point numbers.

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0004
			For sub robot	0x0084
n+2	Command flag	bit 0	(0:Fixed)	0
		bit 2–bit 1	Speed designation flag	bb
		bit 4–bit 3	(0:Fixed)	0
		bit 5	Acceleration designation flag	d
		bit 6	Deceleration designation flag	e
		bit 14–bit 7	(0:Fixed)	0
		bit 15	Current position output designation flag	n
n+4	Not used			0x0000
n+6	Specified speed			0xssss
n+8	First point number			0xpppp
n+10	Second point number			0xpppp
n+12 to n+18	Not used			0x0000
n+20	Acceleration designation			0xrrrr
n+22	Deceleration designation			0xrrrr
n+24 to n+30	Not used			0x0000

bb :Specify the speed setting method in 2 bits.

Value	Meaning
00	Speed is not specified.
10	Speed is set in %.
11	Speed is specified in mm/s.

d :Specify in 1 bit whether to set acceleration.

Value	Meaning
0	Acceleration is not specified.
1	Acceleration is specified.

e :Specifies in 1 bit whether to set deceleration.

Value	Meaning
0	Deceleration is not specified.
1	Deceleration is specified.

n :Specify in 1 bit whether to output current position.

Value	Meaning
0	No output.
1	Output.

ssss :Specify the speed in 16 bits.
Specified range: Speed % : 1 (=0x0001) to 100 (=0x0064)
Specified speed in mm/s : 1 (=0x0001) to 1000 (=0x03E8)

pppp :Specify the first and second point numbers in 16 bits.
Specified range: 0 (=0x0000) to 9999 (=0x270F)

rrrr :Specify the acceleration and deceleration in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data. Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Specify the MOVE command with circular interpolation as shown at right, when moving all axes of the main robot to point numbers 100 and 101 at 20% speed and 50% deceleration. The current position output is specified at this time.

Channel	Value
n	0x0004
n+2	0x8044
n+4	0x0000
n+6	0x0014
n+8	0x0064
n+10	0x0065
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0032
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when controller display units are in millimeters and with:

Axis 1 = 123.45
Axis 2 = -1.23
Axis 3 = 50.00
Axis 4 = 90.23
Other axes = 0.00

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0001
m+8	0x3039
m+10	0x0000
m+12	0xFF85
m+14	0xFFFF
m+16	0x1388
m+18	0x0000
m+20	0x233F
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2.1.5 Direct PTP designation (millimeter units)

This command moves the robot to a target position in PTP motion by directly specifying the data in millimeters.

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0006
			For sub robot	0x0086
n+2	Command flag	bit 0	Axis designation flag	a
		bit 2–bit 1	Speed designation flag	bb
		bit 14–bit 3	(0:Fixed)	0
		bit 15	Current position output designation flag	n
n+4	Specified axis to move	bit 0	Axis 1	0x00tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 15–bit 6	(0:Fixed)	
n+6	Specified speed			0xssss
n+8	Axis-1 data			0xpppppppp
n+10				
n+12	Axis-2 data			0xpppppppp
n+14				
n+16	Axis-3 data			0xpppppppp
n+18				
n+20	Axis-4 data			0xpppppppp
n+22				
n+24	Axis-5 data			0xpppppppp
n+26				
n+28	Axis-6 data			0xpppppppp
n+30				

a :Specify in 1 bit whether all axes are designated.

Value	Meaning
0	All axes are specified.
1	One or more axes are specified.

bb :Specify the speed setting method in 2 bits.

Value	Meaning
00	Speed is not specified.
10	Speed is set in %.

n :Specify in 1 bit whether to output current position.

Value	Meaning
0	No output.
1	Output.

tt :Specify the axis to move in bit pattern using lower 8 bits. Valid when axis designation flag is 1.

ssss :Specify the speed in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

pppppppp :Specify the target position data for each axis in 32 bits. (little endian)
Data should be integers (x100) in millimeter units.

CAUTION

Do not try to specify only axis 4 to move on SCARA robots. Attempting to specify only axis 4 will cause a "5.38: Illegal option".
When specifying axis 4, then also specify axis 1 and axis 2 at the same time.

Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data. Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Specify the MOVE command with direct PTP designation (millimeter units) as shown at right, when moving all axes of the main robot to the following points at 50% speed.

Axis 1 = 100.00

Axis 2 = -200.00

Axis 3 = 50.00

Axis 4 = -180.00

Other axes = 0.00

The current position output is specified at this time.

Channel	Value
n	0x0006
n+2	0x8004
n+4	0x0000
n+6	0x0032
n+8	0x2710
n+10	0x0000
n+12	0xB1E0
n+14	0xFFFF
n+16	0x1388
n+18	0x0000
n+20	0xB9B0
n+22	0xFFFF
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right, when controller display units are in millimeters.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0001
m+8	0x2710
m+10	0x0000
m+12	0xB1E0
m+14	0xFFFF
m+16	0x1388
m+18	0x0000
m+20	0xB9B0
m+22	0xFFFF
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2.1.6 Direct PTP designation (pulse units)

This command moves the robot to a target position in PTP motion by directly specifying the data in pulses.

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0007
			For sub robot	0x0087
n+2	Command flag	bit 0	Axis designation flag	a
		bit 2–bit 1	Speed designation flag	bb
		bit 14–bit 3	(0:Fixed)	0
		bit 15	Current position output designation flag	n
n+4	Specified axis to move	bit 0	Axis 1	0x00tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 15–bit 6	(0:Fixed)	
n+6	Specified speed			0xssss
n+8	Axis-1 data			0xpppppppp
n+10				
n+12	Axis-2 data			0xpppppppp
n+14				
n+16	Axis-3 data			0xpppppppp
n+18				
n+20	Axis-4 data			0xpppppppp
n+22				
n+24	Axis-5 data			0xpppppppp
n+26				
n+28	Axis-6 data			0xpppppppp
n+30				

a :Specify in 1 bit whether all axes are designated.

Value	Meaning
0	All axes are specified.
1	One or more axes are specified.

bb :Specify the speed setting method in 2 bits.

Value	Meaning
00	Speed is not specified.
10	Speed is set in %.

n :Specify in 1 bit whether to output current position.

Value	Meaning
0	No output.
1	Output.

tt :Specify the axis to move in bit pattern using lower 8 bits.
Valid when axis designation flag is 1.

ssss :Specify the speed in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

pppppppp :Specify the target position data for each axis in 32 bits. (little endian)
Data should be integers in pulse units.

Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data. Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Specify the MOVE command with direct designation PTP (pulse units) as shown at right, when moving all axes of the main robot to the following points at 50% speed.

Axis 1 = 100000
Axis 2 = -200000
Axis 3 = 50000
Axis 4 = -180000
Other axes = 0

The current position output is specified at this time.

Channel	Value
n	0x0007
n+2	0x8004
n+4	0x0000
n+6	0x0032
n+8	0x86A0
n+10	0x0001
n+12	0xF2C0
n+14	0xFFFC
n+16	0xC350
n+18	0x0000
n+20	0x40E0
n+22	0xFFFD
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right, when controller display units are in pulses.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0x86A0
m+10	0x0001
m+12	0xF2C0
m+14	0xFFFC
m+16	0xC350
m+18	0x0000
m+20	0x40E0
m+22	0xFFFD
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2.2 MOVEI command

Execute this command group to move the robot to a relative position.

4.2.2.1 PTP designation

This command moves the robot a specified distance in PTP motion by specifying the point number.

NOTE

- If the MOVEI command is interrupted and then re-executed, the resumed motion that occurs either to the original target position or to a new target position referenced to the current position can be selected by the "MOVEI/DRIVEI start position" setting of other parameters. For details, refer to the controller user's manual.
- The other parameters default "MOVEI/DRIVEI start position" setting is Keep (motion to the original target position when MOVEI is interrupted and then re-executed).

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0009
			For sub robot	0x0089
n+2	Command flag	bit 0	Axis designation flag	a
		bit 2–bit 1	Speed designation flag	bb
		bit 14–bit 3	(0:Fixed)	0
		bit 15	Current position output designation flag	n
n+4	Specified axis to move	bit 0	Axis 1	0x00tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 15–bit 6	(0:Fixed)	
n+6	Specified speed			0xssss
n+8	Point number			0xpppp
n+10	Not used			0x0000
to				
n+30				

a :Specify in 1 bit whether all axes are designated.

Value	Meaning
0	All axes are specified.
1	One or more axes are specified.

bb :Specify the speed setting method in 2 bits.

Value	Meaning
00	Speed is not specified.
10	Speed is set in %.

n :Specify in 1 bit whether to output current position.

Value	Meaning
0	No output.
1	Output.

tt :Specify the axis to move in bit pattern using lower 8 bits.
Valid when axis designation flag is 1.

ssss :Specify the movement speed in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

pppp :Specify the point number in 16 bits.
Specified range: 0 (=0x0000) to 9999 (=0x270F)

Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data.
Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Specify the MOVEI command with PTP designation as shown at right, when moving all axes of the main robot a distance specified by point number 100 at 50% speed. The current position output is specified at this time.

Channel	Value
n	0x0009
n+2	0x8004
n+4	0x0000
n+6	0x0032
n+8	0x0064
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when controller display units are in pulses and with:

Axis 1 = 123456

Axis 2 = -123

Other axes = 0

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0xE240
m+10	0x0001
m+12	0xFF85
m+14	0xFFFF
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2.2.2 Direct PTP designation (millimeter units)

This command moves the robot a specified data distance in PTP motion by directly specifying the data in millimeters.

NOTE

- If the MOVEI command is interrupted and then re-executed, the resumed motion that occurs either to the original target position or to a new target position referenced to the current position can be selected by the "MOVEI/DRIVEI start position" setting of other parameters. For details, refer to the controller user's manual.
- The other parameters default "MOVEI/DRIVEI start position" setting is Keep (motion to the original target position when MOVEI is interrupted and then re-executed).

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x000E
			For sub robot	0x008E
n+2	Command flag	bit 0	Axis designation flag	a
		bit 2–bit 1	Speed designation flag	bb
		bit 14–bit 3	(0:Fixed)	0
		bit 15	Current position output designation flag	n
n+4	Specified axis to move	bit 0	Axis 1	0x00tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 15–bit 6	(0:Fixed)	
n+6	Specified speed			0xssss
n+8	Axis-1 data			0xpppppppp
n+10				
n+12	Axis-2 data			0xpppppppp
n+14				
n+16	Axis-3 data			0xpppppppp
n+18				
n+20	Axis-4 data			0xpppppppp
n+22				
n+24	Axis-5 data			0xpppppppp
n+26				
n+28	Axis-6 data			0xpppppppp
n+30				

a :Specify in 1 bit whether all axes are designated.

Value	Meaning
0	All axes are specified.
1	One or more axes are specified.

bb :Specify the speed setting method in 2 bits.

Value	Meaning
00	Speed is not specified.
10	Speed is set in %.

n :Specify in 1 bit whether to output current position.

Value	Meaning
0	No output.
1	Output.

tt :Specify the axis to move in bit pattern using lower 8 bits.
Valid when axis designation flag is 1.

ssss :Specify the speed in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

pppppppp :Specify the target movement distance data for each axis in 32 bits. (little endian)
Data should be integers (x100) in millimeter units.

Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data.
Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Specify the MOVEI command with direct PTP designation (millimeter units) as shown at right, when moving all axes of the main robot a distance specified by the following points from "0.00" mm positions at 50% speed.

Axis 1 = 100.00

Axis 2 = -200.00

Axis 3 = 50.00

Axis 4 = -180.00

Other axes = 0.00

The current position output is specified at this time.

Channel	Value
n	0x000E
n+2	0x8004
n+4	0x0000
n+6	0x0032
n+8	0x2710
n+10	0x0000
n+12	0xB1E0
n+14	0xFFFF
n+16	0x1388
n+18	0x0000
n+20	0xB9B0
n+22	0xFFFF
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right, when controller display units are in millimeters.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0001
m+8	0x2710
m+10	0x0000
m+12	0xB1E0
m+14	0xFFFF
m+16	0x1388
m+18	0x0000
m+20	0xB9B0
m+22	0xFFFF
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2.2.3 Direct PTP designation (pulse units)

This command moves the robot a specified data distance in PTP motion by directly specifying the data in pulses.

NOTE

- If the MOVEI command is interrupted and then re-executed, the resumed motion that occurs either to the original target position or to a new target position referenced to the current position can be selected by the "MOVEI/DRIVEI start position" setting of other parameters. For details, refer to the controller user's manual.
- The other parameters default "MOVEI/DRIVEI start position" setting is Keep (motion to the original target position when MOVEI is interrupted and then re-executed).

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x000F
			For sub robot	0x008F
n+2	Command flag	bit 0	Axis designation flag	a
		bit 2–bit 1	Speed designation flag	bb
		bit 14–bit 3	(0:Fixed)	0
		bit 15	Current position output designation flag	n
n+4	Specified axis to move	bit 0	Axis 1	0x00tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 15–bit 6	(0:Fixed)	
n+6	Specified speed			0xssss
n+8	Axis-1 data			0xpppppppp
n+10				
n+12	Axis-2 data			0xpppppppp
n+14				
n+16	Axis-3 data			0xpppppppp
n+18				
n+20	Axis-4 data			0xpppppppp
n+22				
n+24	Axis-5 data			0xpppppppp
n+26				
n+28	Axis-6 data			0xpppppppp
n+30				

a :Specify in 1 bit whether all axes are designated.

Value	Meaning
0	All axes are specified.
1	One or more axes are specified.

bb :Specify the speed setting method in 2 bits.

Value	Meaning
00	Speed is not specified.
10	Speed is set in %.

n :Specify in 1 bit whether to output current position.

Value	Meaning
0	No output.
1	Output

tt :Specify the axis to move in bit pattern using lower 8 bits.
Valid when axis designation flag is 1.

ssss :Specify the speed in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

pppppppp :Specify the target movement distance data for each axis in 32 bits. (little endian)
Data should be integers in pulse units.

Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data.
Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Specify the MOVEI command with direct PTP designation (millimeter units) as shown at right, when moving all axes of the main robot a distance specified by the following points from "0" pulse positions at 50% speed.

Axis 1 = 100000
Axis 2 = -200000
Axis 3 = 50000
Axis 4 = -180000
Other axes = 0

The current position output is specified at this time.

Channel	Value
n	0x000F
n+2	0x8004
n+4	0x0000
n+6	0x0032
n+8	0x86A0
n+10	0x0001
n+12	0xF2C0
n+14	0xFFFC
n+16	0xC350
n+18	0x0000
n+20	0x40E0
n+22	0xFFFD
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right, when controller display units are in millimeters.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0x86A0
m+10	0x0001
m+12	0xF2C0
m+14	0xFFFC
m+16	0xC350
m+18	0x0000
m+20	0x40E0
m+22	0xFFFD
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2.3 DRIVE command

Execute this command group to move the specified axis of the robot to an absolute position. Valid only for a single axis.

4.2.3.1 Point designation

This command moves the specified axis of the robot to a target position in PTP motion by specifying the point number.

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0010
			For sub robot	0x0090
n+2	Command flag	bit 0	(1:Fixed)	1
		bit 2–bit 1	Speed designation flag	bb
		bit 6–bit 3	(0:Fixed)	0
		bit 7	Torque limit designation flag	h
		bit 14–bit 8	(0:Fixed)	0
		bit 15	Current position output designation flag	n
n+4	Specified axis to move	bit 0	Axis 1	0x00tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 15–bit 6	(0:Fixed)	
n+6	Specified speed			0xssss
n+8	Point number			0xpppp
n+10	Not used			0x0000
to				
n+14				
n+16	Specified torque			0xqqqq
n+18	Not used			0x0000
to				
n+30				

bb :Specify the speed setting method in 2 bits.

Value	Meaning
00	Speed is not specified.
10	Speed is set in %.

h :Specify in 1 bit whether to use torque limit.

Value	Meaning
0	Torque limit is not specified.
1	Torque limit is specified.

n :Specify in 1 bit whether to output current position.

Value	Meaning
0	No output.
1	Output.

tt :Specify the axis to move in bit pattern using lower 8 bits.
Only one axis can be specified.

ssss :Specify the movement speed in 16 bits.
Specified range: 1 (= 0x0001) to 100 (=0x0064)

pppp :Specify the point number in 16 bits.
Specified range: 0 (=0x0000) to 9999 (=0x270F)

qqqq :Specify the percentage of rated torque in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

CAUTION

- **When the torque limit is specified by the command flag, this command ends in the following cases.**
  - **The axis has reached the target position when the time required to move to the target position has elapsed.**
  - **Time-out period was exceeded while the axis torque has reached the specified torque value. (Time-out period depends on the TRQTIME statement or TRQTIME2 statement executed in the program.)**
- **This command cannot use a torque offset value.**

NOTE

Refer to the programming manual for detailed information on the TRQTIME and TRQTIME2 statements of the robot language.

Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data.
Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Specify the DRIVE command with point designation as shown at right, to move axis 3 of the main robot to point number 100 at 50% speed. The current position output is specified at this time.

Channel	Value
n	0x0010
n+2	0x8005
n+4	0x0004
n+6	0x0032
n+8	0x0064
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right, when controller display units are in pulses and with:

Axis 1 = 123456

Axis 2 = -123

Other axes = 0

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0xE240
m+10	0x0001
m+12	0xFF85
m+14	0xFFFF
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2.3.2 Direct designation (millimeter units)

This command moves the specified axis of the robot to a target position in PTP motion by directly specifying the data in millimeters.

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0012
			For sub robot	0x0092
n+2	Command flag	bit 0	(1:Fixed)	1
		bit 2–bit 1	Speed designation flag	bb
		bit 6–bit 3	(0:Fixed)	0
		bit 7	Torque limit designation flag	h
		bit 14–bit 8	(0:Fixed)	0
		bit 15	Current position output designation flag	n
n+4	Specified axis to move	bit 0	Axis 1	0x00tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 15–bit 6	(0:Fixed)	
n+6	Specified speed			0xssss
n+8	Movement data			0xpppppppp
n+10				
n+12	Not used			0x0000
n+14				
n+16	Not used			0xqqqq
n+18	Not used			0x0000
to				
n+30				

bb :Specify the speed setting method in 2 bits.

Value	Meaning
00	Speed is not specified.
10	Speed is set in %.

h :Specify in 1 bit whether to use torque limit.

Value	Meaning
0	Torque limit is not specified.
1	Torque limit is specified.

n :Specify in 1 bit whether to output current position.

Value	Meaning
0	No output.
1	Output.

tt :Specify the axis to move in bit pattern using lower 8 bits.
Only one axis can be specified.

ssss :Specify the movement speed in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

pppppppp :Specify target position data for specified axis in 32 bits. (little endian)
Data should be integers (x 100) in millimeter units.

qqqq :Specify the percentage of rated torque in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

CAUTION

- **When the torque limit is specified by the command flag, this command ends in the following cases.**
  - **The axis has reached the target position when the time required to move to the target position has elapsed.**
  - **Time-out period was exceeded while the axis torque has reached the specified torque value. (Time-out period depends on the TRQTIME statement or TRQTIME2 statement executed in the program.)**
- **This command cannot use a torque offset value.**

NOTE

Refer to the programming manual for detailed information on the TRQTIME and TRQTIME2 statements of the robot language.

Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data.
Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Specify the DRIVE command with direct designation (millimeter units) as shown at right, to move axis 3 of the main robot to a position of "50.00" at 50% speed. The current position output is specified at this time.

Channel	Value
n	0x0012
n+2	0x8005
n+4	0x0004
n+6	0x0032
n+8	0x1388
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when controller display units are in millimeters and with:

Axis 1 = 100.00
Axis 2 = -200.00
Axis 3 = 50.00
Axis 4 = -180.00
Other axes = 0.00

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0001
m+8	0x2710
m+10	0x0000
m+12	0xB1E0
m+14	0xFFFF
m+16	0x1388
m+18	0x0000
m+20	0xB9B0
m+22	0xFFFF
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2.3.3 Direct designation (pulse units)

This command moves the specified axis of the robot to a target position in PTP motion by directly specifying the data in pulses.

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0013
			For sub robot	0x0093
n+2	Command flag	bit 0	(1:Fixed)	1
		bit 2–bit 1	Speed designation flag	bb
		bit 6–bit 3	(0:Fixed)	0
		bit 7	Torque limit designation flag	h
		bit 14–bit 8	(0:Fixed)	0
		bit 15	Current position output designation flag	n
n+4	Specified axis to move	bit 0	Axis 1	0x00tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 15–bit 6	(0:Fixed)	
n+6	Specified speed			0xssss
n+8	Movement data			0xpppppppp
n+10				
n+12	Not used			0x0000
n+14				
n+16	Specified torque			0xqqqq
n+18	Not used			0x0000
to				
n+30				

bb :Specify the speed setting method in 2 bits.

Value	Meaning
00	Speed is not specified.
10	Speed is set in %.

h :Specify in 1 bit whether to use torque limit.

Value	Meaning
0	Torque limit is not specified.
1	Torque limit is specified.

n :Specify in 1 bit whether to output current position.

Value	Meaning
0	No output.
1	Output.

tt :Specify the axis to move in bit pattern using lower 8 bits.
Only one axis can be specified.

ssss :Specify the movement speed in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

pppppppp :Specify the target position data for specified axis in 32 bits. (little endian)
Data should be integers in pulse units.

qqqq :Specify the percentage of rated torque in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

CAUTION

- **When the torque limit is specified by the command flag, this command ends in the following cases.**
  - **The axis has reached the target position when the time required to move to the target position has elapsed.**
  - **Time-out period was exceeded while the axis torque has reached the specified torque value. (Time-out period depends on the TRQTIME statement or TRQTIME2 statement executed in the program.)**
- **This command cannot use a torque offset value.**

NOTE

Refer to the programming manual for detailed information on the TRQTIME and TRQTIME2 statements of the robot language.

Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data.
Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Specify the DRIVE command with direct designation (pulse units) as shown at right, to move axis 3 of the main robot to a position of "5000" pulses at 50% speed. The current position output is specified at this time.

Channel	Value
n	0x0013
n+2	0x8005
n+4	0x0004
n+6	0x0032
n+8	0x1388
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when controller display units are in pulses and with:

Axis 1 = 10000
Axis 2 = -20000
Axis 3 = 5000
Axis 4 = -18000
Other axes = 0

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0x2710
m+10	0x0000
m+12	0xB1E0
m+14	0xFFFF
m+16	0x1388
m+18	0x0000
m+20	0xB9B0
m+22	0xFFFF
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2.4 DRIVEI command

Execute this command group to move the specified axis of the robot to a relative position. Valid only for a single axis.

4.2.4.1 Point designation

This command moves the specified axis of the robot in PTP motion a distance by specifying the point number.

NOTE

- If the DRIVEI command is interrupted and then re-executed, the resumed motion that occurs either to the original target position or to a new target position referenced to the current position can be selected by the "MOVEI/DRIVEI start position" setting of other parameters. For details, refer to the controller user's manual.
- The other parameters default "MOVEI/DRIVEI start position" setting is Keep (motion to the original target position when DRIVEI is interrupted and then re-executed).

■ **Command**

Channel	Contents			Value
n	Command code		For main robot	0x0014
			For sub robot	0x0094
n+2	Command flag	bit 0	(1:Fixed)	1
		bit 2–bit 1	Speed designation flag	bb
		bit 14–bit 3	(0:Fixed)	0
		bit 15	Current position output designation flag	n
n+4	Specified axis to move	bit 0	Axis 1	0x00tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 15–bit 6	(0:Fixed)	
n+6	Specified speed			0xssss
n+8	Point number			0xpppp
n+10	Not used			0x0000
to				
n+30				

bb :Specify the speed setting method in 2 bits.

Value	Meaning
00	Speed is not specified.
10	Speed is set in %.

n :Specify in 1 bit whether to output current position.

Value	Meaning
0	No output.
1	Output.

tt :Specify the axis to move in bit pattern using lower 8 bits.
Only one axis can be specified.

ssss :Specify the movement speed in 16 bits.
Specified range: 1 (= 0x0001) to 100 (=0x0064)

pppp :Specify the point number in 16 bits.
Specified range: 0 (=0x0000) to 9999 (=0x270F)

Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data.
Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Specify the DRIVEI command with point designation as shown at right, to move axis 3 of the main robot a distance specified by point number 100 at 50% speed. The current position output is specified at this time.

Channel	Value
n	0x0014
n+2	0x8005
n+4	0x0004
n+6	0x0032
n+8	0x0064
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when controller display units are in pulses and with:

Axis 1 = 123456

Axis 2 = -123

Other axes = 0

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0xE240
m+10	0x0001
m+12	0xFF85
m+14	0xFFFF
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2.4.2 Direct designation (millimeter units)

This command moves the specified axis of the robot in PTP motion a distance by directly specifying the data in millimeters.

NOTE

- If the DRIVEI command is interrupted and then re-executed, the resumed motion that occurs either to the original target position or to a new target position referenced to the current position can be selected by the "MOVEI/DRIVEI start position" setting of other parameters. For details, refer to the controller user's manual.
- The other parameters default "MOVEI/DRIVEI start position" setting is Keep (motion to the original target position when DRIVEI is interrupted and then re-executed).

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0016
			For sub robot	0x0096
n+2	Command flag	bit 0	(1:Fixed)	1
		bit 2–bit 1	Speed designation flag	bb
		bit 14–bit 3	(0:Fixed)	0
		bit 15	Current position output designation flag	n
n+4	Specified axis to move	bit 0	Axis 1	0x00tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 15–bit 6	(0:Fixed)	
n+6	Specified speed			0xssss
n+8	Movement data			0xpppppppp
n+10				
n+12	Not used			0x0000
to				
n+30				

bb :Specify the speed setting method in 2 bits.

Value	Meaning
00	Speed is not specified.
10	Speed is set in %.

n :Specify in 1 bit whether to output current position.

Value	Meaning
0	No output.
1	Output.

tt :Specify the axis to move in bit pattern using lower 8 bits.
Only one axis can be specified.

ssss :Specify the speed in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

pppppppp :Specify the target movement distance data for specified axis in 32 bits. (little endian)
Data should be integers (x100) in millimeter units.

Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data.
Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Specify the DRIVEI command with direct designation (millimeter units) as shown at right, to move axis 3 a distance equal to "50.00" from "0.00" position at 50% speed. The current position output is specified at this time.

Channel	Value
n	0x0016
n+2	0x8005
n+4	0x0004
n+6	0x0032
n+8	0x1388
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when controller display units are in millimeters and with:

Axis 1 = 100.00

Axis 2 = -200.00

Axis 3 = 50.00

Axis 4 = -180.00

Other axes = 0.00

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0001
m+8	0x2710
m+10	0x0000
m+12	0xB1E0
m+14	0xFFFF
m+16	0x1388
m+18	0x0000
m+20	0xB9B0
m+22	0xFFFF
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2.4.3 Direct designation (pulse units)

This command moves the specified axis of the robot in PTP motion a distance by directly specifying the data in pulses.

NOTE

- If the DRIVEI command is interrupted and then re-executed, the resumed motion that occurs either to the original target position or to a new target position referenced to the current position can be selected by the "MOVEI/DRIVEI start position" setting of other parameters. For details, refer to the controller user's manual.
- The other parameters default "MOVEI/DRIVEI start position" setting is Keep (motion to the original target position when DRIVEI is interrupted and then re-executed).

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0017
			For sub robot	0x0097
n+2	Command flag	bit 0	(1:Fixed)	1
		bit 2–bit 1	Speed designation flag	bb
		bit 14–bit 3	(0:Fixed)	0
		bit 15	Current position output designation flag	n
n+4	Specified axis to move	bit 0	Axis 1	0x00tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 15–bit 6	(0:Fixed)	
n+6	Specified speed			0xssss
n+8	Movement data			0xpppppppp
n+10				
n+12	Not used			0x0000
to				
n+30				

bb :Specify the speed setting method in 2 bits.

Value	Meaning
00	Speed is not specified.
10	Speed is set in %.

n :Specify in 1 bit whether to output current position.

Value	Meaning
0	No output.
1	Output.

tt :Specify the axis to move in bit pattern using lower 8 bits.
Only one axis can be specified.

ssss :Specify the movement speed in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

pppppppp :Specify the target movement distance data for specified axis in 32 bits. (little endian)
Data should be integers in pulse units.

Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data.
Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Specify the DRIVEI command with direct designation (pulse units) as shown at right, to move axis 3 a distance equal to "5000" pulses from "0" pulse position at 50% speed. The current position output is specified at this time.

Channel	Value
n	0x0017
n+2	0x8005
n+4	0x0004
n+6	0x0032
n+8	0x1388
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when controller display units are in pulses and with:

Axis 1 = 10000
Axis 2 = -20000
Axis 3 = 5000
Axis 4 = -18000
Other axes = 0

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0x2710
m+10	0x0000
m+12	0xB1E0
m+14	0xFFFF
m+16	0x1388
m+18	0x0000
m+20	0xB9B0
m+22	0xFFFF
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2.5 Pallet movement command

Execute this command group to move the robot to work positions on a pallet.

4.2.5.1 PTP designation

This command moves the robot to a target position in PTP motion by specifying the pallet number and work position number.

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0018
			For sub robot	0x0098
n+2	Command flag	bit 0	(0:Fixed)	0
		bit 2–bit 1	Speed designation flag	bb
		bit 14–bit 3	(0:Fixed)	0
		bit 15	Current position output designation flag	n
n+4	Not used			0x0000
n+6	Specified speed			0xssss
n+8	Pallet number			0xpppp
n+10	Work position number			0xwwww
n+12	Not used			0x0000
to				
n+30				

bb :Specify the speed setting method in 2 bits.

Value	Meaning
00	Speed is not specified.
10	Speed is set in %.

n :Specify in 1 bit whether to output current position.

Value	Meaning
0	No output.
1	Output.

ssss :Specify the movement speed in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

pppp :Specify the pallet number in 16 bits.
Specified range: 0 (=0x0000) to 19 (=0x0013)

wwww :Specify the work position number in 16 bits.
Specified range: 1 (=0x0001) to 32767 (=0x7FFF)

Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data.
Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Specify the PMOVE command with PTP designation as shown at right, when moving the main robot to work position number 21 on pallet number 1 at 70% speed. The current position output is specified at this time.

Channel	Value
n	0x0018
n+2	0x8004
n+4	0x0000
n+6	0x0046
n+8	0x0001
n+10	0x0015
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when controller display units are in millimeters and with:

Axis 1 = 123.45

Axis 2 = -1.23

Axis 3 = 20.00

Other axes = 0.00

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0001
m+8	0x3039
m+10	0x0000
m+12	0xFF85
m+14	0xFFFF
m+16	0x07D0
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2.5.2 Arch designation

This command moves the robot to a target position in arch motion by specifying the pallet number, work position number, arch axis and arch data.

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0019
			For sub robot	0x0099
n+2	Command flag	bit 0	(0:Fixed)	0
		bit 2–bit 1	Speed designation flag	bb
		bit 3	(0:Fixed)	0
		bit 4	Arch data unit flag	d
		bit 14–bit 5	(0:Fixed)	0
		bit 15	Current position output designation flag	n
n+4	Arch designation axis	bit 7–bit 0	(0:Fixed)	0xuu00
		bit 8	Axis 1	
		bit 9	Axis 2	
		bit 10	Axis 3	
		bit 11	Axis 4	
		bit 12	Axis 5	
		bit 13	Axis 6	
		bit 15–bit 14	(0:Fixed)	
n+6	Specified speed			0xssss
n+8	Pallet number			0xpppp
n+10	Work position number			0xwwww
n+12	Not used			0x0000
n+14				
n+16	Arch position data			0xqqqqqqqq
n+18				
n+20	Not used			0x0000
to				
n+30				

bb :Specify the speed setting method in 2 bits.

Value	Meaning
00	Speed is not specified.
10	Speed is set in %.

d :Specify the arch data units in 1 bit.

Value	Meaning
0	Pulse units
1	Millimeter units

n :Specify in 1 bit whether to output current position.

Value	Meaning
0	No output.
1	Output.

uu :Specify the arch motion axis in bit pattern using upper 8 bits.
Specified arch axis is one axis only.

ssss :Specify the speed in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

pppp :Specify the pallet number in 16 bits.
Specified range: 0 (=0x0000) to 19 (=0x0013)

wwww :Specify the work position number in 16 bits.
Specified range: 1 (=0x0001) to 32767 (=0x7FFF)

qqqqqqqq :Specify the arch position data in 32 bits. (little endian)
Data should be integers when units are in pulses.
Data should be integers (x100) when units are in millimeters.

■ Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data.
Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.
ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Specify the PMOVE command with arch designation as shown at right, when moving the Z-axis to work position number 32 on pallet number 10 at 70% speed by way of a Z-axis arch position of 10.00mm. The current position output is specified at this time.

Channel	Value
n	0x0019
n+2	0x8014
n+4	0x0400
n+6	0x0046
n+8	0x000A
n+10	0x0020
n+12	0x0000
n+14	0x0000
n+16	0x03E8
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when controller display units are in millimeters and with:

Axis 1 = 123.45
Axis 2 = -1.23
Axis 3 = 50.00
Axis 4 = 90.23
Other axes = 0.00

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0001
m+8	0x3039
m+10	0x0000
m+12	0xFF85
m+14	0xFFFF
m+16	0x1388
m+18	0x0000
m+20	0x233F
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2.6 Jog movement command

Execute this command in MANUAL mode to move the robot in jog mode. This command can only be executed in MANUAL mode. This command is linked with the controller's point display unit. So the axis moves in PTP motion when the display units are in pulses, and moves by linear interpolation on the Cartesian coordinates when the units are in millimeters. Jog speed is determined by the manual movement speed.

To stop the jog command, set the dedicated input of the interlock signal (SI11) to OFF. Abnormal end status (0x4000) appears as the status code and the error code indicates that the robot has stopped by the interlock (0x0C02). After confirming that movement has stopped, set the dedicated input of the interlock signal to ON.

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0020
			For sub robot	0x00A0
n+2	Command flag	bit 14–bit 0	(0:Fixed)	0
		bit 15	Current position output designation flag	n
n+4	Axis to move and direction	bit 0	Axis 1	tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 6	(0:Fixed)	0
		bit 7	Direction	d
		bit 15–bit 8	(0:Fixed)	0
n+6	Not used			0x0000
to				
n+30				

n :Specify in 1 bit whether to output current position.

Value	Meaning
0	No output.
1	Output.

tt :Specify the axis to move in 0 to 3 bits.
Only one axis can be specified.

d :Specify the movement direction in 1 bit.

Value	Meaning
0	+ direction
1	– direction

Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data.
Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end (When jog movement was stopped by interlock)

Channel	Contents			Value
m	Status code			0x4000
m+2	Error code			0x0C02
m+4	Not used			
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data.
Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end (other cases)

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Specify the jog command as shown at right, to move axis 1 of the main robot in the minus (-) direction. The current position output is specified at this time.

Channel	Value
n	0x0020
n+2	0x8000
n+4	0x0081
n+6	0x0000
n+8	0x0000
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right, after robot movement with the jog command is stopped by the interlock signal, when controller display units are in millimeters and with:

Axis 1 = 123.45

Axis 2 = -1.23

Axis 3 = 20.00

Other axes = 0.00

Channel	Value
m	0x4000
m+2	0x0C02
m+4	0x0000
m+6	0x0001
m+8	0x3039
m+10	0x0000
m+12	0xFF85
m+14	0xFFFF
m+16	0x07D0
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2.7 Inching movement command

Execute this command to move the robot by inching in MANUAL mode. Inching movement distance is linked to the manual movement speed. The inching command can only be executed in MANUAL mode.
This command is linked with the controller's point display unit. So when display units are in pulses, the axis moves a certain number of pulses at the manual speed setting. When display units are in millimeters, the axis moves on Cartesian coordinates by linear interpolation at the manual speed setting divided by 100.

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0024
			For sub robot	0x00A4
n+2	Command flag	bit 14–bit 0	(0:Fixed)	0
		bit 15	Current position output designation flag	n
n+4	Axis to move and direction	bit 0	Axis 1	tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 6	(0:Fixed)	0
		bit 7	Direction	d
		bit 15–bit 8	(0:Fixed)	0
n+6	Not used			0x0000
to				
n+30				

n :Specify in 1 bit whether to output current position.

Value	Meaning
0	No output.
1	Output.

tt :Specify the axis to move in 0 to 3 bits.
Only one axis can be specified.

d :Specify the movement direction in 1 bit.

Value	Meaning
0	+ direction
1	- direction

Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	a
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

a :Shows in 1 bit the units for current position output point data.
Linked with point display unit of controller.

Value	Meaning
0	Pulse units
1	Millimeter units

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers when point display units are in pulses.
Data is shown in integers (x100) when point display units are in millimeters.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Specify the inching command as shown at right, to move axis 2 of the main robot in the plus direction. When controller's point display units are in millimeters, movement is in 0.5 mm steps at 50% manual speed. The current position output is specified at this time.

Channel	Value
n	0x0024
n+2	0x8000
n+4	0x0002
n+6	0x0000
n+8	0x0000
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right, after executing the inching command and then stopping point movement, when controller display units are in millimeters and with:

Axis 1 = 123.45

Axis 2 = -1.23

Axis 3 = 20.00

Other axes = 0.00

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0001
m+8	0x3039
m+10	0x0000
m+12	0xFF85
m+14	0xFFFF
m+16	0x07D0
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.2.8 Point teaching command

Execute this command to teach the current robot position to the specified point number.
Point data units of this command are linked to the controller's point display unit.

■ Command

Channel	Contents		Value
n	Command code	For main robot	0x0028
		For sub robot	0x00A8
n+2	Not used		0x0000
n+4	Point number		0xpppp
n+6	Not used		0x0000
to			
n+30			

pppp :Specify the point number in 16 bits.
Specified range: 0 (= 0x0000) to 9999 (=0x270F)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.
ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the point teaching command as shown at right, to teach the main robot current position to point 4000.

Channel	Value
n	0x0028
n+2	0x0000
n+4	0x0FA0
n+6	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
to	
m+30	

4.2.9 Absolute reset movement command

When absolute reset of the specified axis uses the mark method, this command moves the axis to the nearest position where absolute reset can be executed. Positions capable of absolute reset are located at every 1/4 rotation of the motor.

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0030
			For sub robot	0x00B0
n+2	Not used			0x0000
n+4	Axis to move and direction	bit 0	Axis 1	tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 6	(0:Fixed)	0
		bit 7	Direction	d
		bit 15–bit 8	(0:Fixed)	0
n+6	Not used			0x0000
to				
n+30				

tt :Specify the axis to move in 0 to 3 bits.
Only one axis can be specified.

d :Specify the movement direction in 1 bit.

Value	Meaning
0	+ direction
1	- direction

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the absolute reset movement command as shown at right, to move axis 2 of the main robot in the minus (-) direction to a position capable of absolute reset.

Channel	Value
n	0x0030
n+2	0x0000
n+4	0x0082
n+6	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
to	
m+30	

4.2.10 Absolute reset command

Execute this command to perform absolute reset on the specified axis. When absolute reset of the specified axis uses the mark method, a position capable of absolute reset is required. If no particular axis is specified (n+4 is 0), then absolute reset is performed on all axes (main robot + sub robot) in either case of command code 0x0031 or 0x00B1. This command cannot be executed if return-to-origin on a mark-specified axis is incomplete. Perform absolute reset on each axis.

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0031
			For sub robot	0x00B1
n+2	Not used			0x0000
n+4	Specified axis	bit 0	Axis 1	0x00tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 15–bit 6	(0:Fixed)	
n+6	Not used			0x0000
to				
n+30				

tt :pecify the axis to perform absolute reset in 0 to 3 bits.
Only one axis can be specified.
If no particular axis is specified then absolute reset is performed on all axes (main robot + sub robot).

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to perform absolute reset on axis 2 of the main robot.

Channel	Value
n	0x0031
n+2	0x0000
n+4	0x0002
n+6	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
to	
m+30	

4.2.11 Return-to-origin command

This command executes return-to-origin on the specified axis. When this command is executed on an incremental mode axis, that axis moves to its origin. When executed on a semi-absolute mode axis, an absolute search is performed on that axis. If no axis is specified (n+4 is 0), return-to-origin is performed on all axes (main robot + sub robot) in either case of command code 0x0032 or 0x00B2.

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0032
			For sub robot	0x00B2
n+2	Not used			0x0000
n+4	Specified axis	bit 0	Axis 1	0x00tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 15-bit 6	(0:Fixed)	
n+6	Not used			0x0000
to				
n+30				

tt :Specify the axis to perform return-to-origin in 0 to 3 bits.
Only one axis can be specified.
If no particular axis is specified then return-to-origin is performed on all axes (main robot + sub robot).

Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to perform return-to-origin on axis 2 of the main robot.

Channel	Value
n	0x0032
n+2	0x0000
n+4	0x0002
n+6	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
to	
m+30	

4.2.12 Servo command

Execute this command group to operate the robot servos.

Servo ON :

Execute this command to turn on the servo of a specified axis. The motor power must be turned on when specifying the axis. All controller servos are turned on if no axis is specified.

Servo OFF :

Execute this command to turn off the servo of a specified axis. All controller servos are turned off if no axis is specified.

Servo Free :

Execute this command to turn off the mechanical brake and dynamic brake after turning off the servo of a specified axis. Servo OFF and Free are repeated when this command is consecutively executed.

Power ON:

Execute this command to turn on the motor power. No axis can be specified.

■ Command

Channel	Contents			Value
n	Command code	Servo ON	For main robot	0x0034
			For sub robot	0x00B4
		Servo OFF	For main robot	0x0035
			For sub robot	0x00B5
		Servo Free	For main robot	0x0036
			For sub robot	0x00B6
		Power ON	All controller servos	0x0037
n+2	Not used			0x0000
n+4	Specified axis	bit 0	Axis 1	0x00tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 15–bit 6	(0:Fixed)	
n+6	Not used			0x0000
to				
n+30				

tt :Specify the axis to move in 0 to 3 bits. All controller servos are processed if no axis is specified. No axis can be specified when executing Power ON.

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the servo command as shown at right, to free the servo of axis 4 of the main robot.

Channel	Value
n	0x0036
n+2	0x0000
n+4	0x0008
n+6	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
to	
m+30	

4.2.13 Manual speed change command

Execute this command to change the manual movement speed in MANUAL mode. This command can only be executed in MANUAL mode.

■ Command

Channel	Contents		Value
n	Command code	For main robot	0x0038
		For sub robot	0x00B8
n+2	Not used		0x0000
n+4	Specified speed		0xssss
n+6	Not used		0x0000
to			
n+30			

ssss :Specify the manual movement speed in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the manual speed change command as shown at right, to set the manual movement speed of the main robot to 20%.

Channel	Value
n	0x0038
n+2	0x0000
n+4	0x0014
n+6	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
to	
m+30	

4.2.14 Auto speed change command

Execute this command to change the auto movement speed in AUTO mode.

■ Command

Channel	Contents		Value
n	Command code	For main robot	0x0039
		For sub robot	0x00B9
n+2	Not used		0x0000
n+4	Specified speed		0xssss
n+6	Not used		0x0000
to			
n+30			

ssss :Specify the auto movement speed in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the auto speed change command as shown at right, to set the auto movement speed of the main robot to 80%.

Channel	Value
n	0x0039
n+2	0x0000
n+4	0x0050
n+6	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
to	
m+30	

4.2.15 Program speed change command

Execute this command to change the program speed in AUTO mode. The program speed changed with this command is reset to 100% when the program is reset or changed.

■ Command

Channel	Contents		Value
n	Command code	For main robot	0x003A
		For sub robot	0x00BA
n+2	Not used		0x0000
n+4	Specified speed		0xssss
n+6	Not used		0x0000
to			
n+30			

ssss :Specify the program speed in 16 bits.
Specified range: 1 (=0x0001) to 100 (=0x0064)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the program speed change command as shown at right, to set the program speed for the main robot to 80%.

Channel	Value
n	0x003A
n+2	0x0000
n+4	0x0050
n+6	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
to	
m+30	

4.2.16 Shift designation change command

Execute this command to change the selected shift to a specified shift number.

■ Command

Channel	Contents		Value
n	Command code	For main robot	0x003B
		For sub robot	0x00BB
n+2	Not used		0x0000
n+4	Specified shift number		0xssss
n+6	Not used		0x0000
to			
n+30			

ssss :Specify the shift number in 16 bits.
Specified range: 0 (=0x0000) to 9 (0x0009)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the shift designation change command as shown at right, to set the shift number of the main robot to shift 4.

Channel	Value
n	0x003B
n+2	0x0000
n+4	0x0004
n+6	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
to	
m+30	

4.2.17 Hand designation change command

Execute this command to change the selected hand to a specified hand number.

■ Command

Channel	Contents		Value
n	Command code	For main robot	0x003C
		For sub robot	0x00BC
n+2	Not used		0x0000
n+4	Specified hand number		0xssss
n+6	Not used		0x0000
to			
n+30			

ssss :Specify the hand number in 16 bits.
Specified range for main robot : 0 (=0x0000) to 3 (0x0003)
Specified range for sub robot : 4 (=0x0004) to 7 (0x0007)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the hand designation change command as shown at right, to set the hand number of the main robot to hand 1.

Channel	Value
n	0x003C
n+2	0x0000
n+4	0x0001
n+6	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
to	
m+30	

4.2.18 Arm designation change command

Execute this command to change the arm designation status. This command is valid only when SCARA robot is specified.

■ Command

Channel	Contents		Value
n	Command code	For main robot	0x003D
		For sub robot	0x00BD
n+2	Not used		0x0000
n+4	Status of specified arm		0xssss
n+6	Not used		0x0000
to			
n+30			

ssss :Specify the arm designation status in 16 bits.

Value	Meaning
0x0000	Right-handed system
0x0001	Left-handed system

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the arm designation change command as shown at right, to set the arm designation status of the main robot to the right-handed system.

Channel	Value
n	0x003D
n+2	0x0000
n+4	0x0000
n+6	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
to	
m+30	

4.2.19 Point display unit designation command

Execute this command to change the point display unit.

■ Command

Channel	Contents	Value
n	Command code	0x003E
n+2	Not used	0x0000
n+4	Display units for specified point	0xssss
n+6	Not used	0x0000
to		
n+30		

ssss :Specify the point display unit system in 16 bits.

Value	Meaning
0x0000	Pulse units
0x0001	Millimeter units
0x0002	Millimeter units (Tool coordinates)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the point display unit designation command as shown at right, to set the point display units to pulses.

Channel	Value
n	0x003E
n+2	0x0000
n+4	0x0000
n+6	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
to	
m+30	

4.3 Category 2 remote commands

Category 2 remote commands are used to define or obtain point data.
A command list is given below.

No.	Command contents			Command code n
1	Point-related commands	Point data definition		0x0100
		Point data reference		0x0101
2	Point comment-related commands	Point comment data definition		0x0104
		Point comment data reference		0x0105
3	Pallet-related command	Pallet data definition		0x0108
		Pallet data reference		0x0109
4	Shift-related command	Shift data definition		0x010C
		Shift data reference		0x010D
5	Hand-related command	Hand data definition	Main robot	0x0110
			Sub robot	0x0190
		Hand data reference		0x0111

m: Start address of the input area assigned to the master module
n : Start address of the output area assigned to the master module
("m" indicates the data direction from remote to master, and "n" the data direction from master to remote.)

4.3.1 Point-related command

Execute this command to define or obtain point data.

4.3.1.1 Point data definition

This command defines point data by specifying the point number and position data on each axis.

■ Command

Channel	Contents			Value
n	Command code			0x0100
n+2	Command flag	bit 0	Point unit	u
		bit 2–bit 1	Hand system	tt
		bit 15–bit 3	(0:Fixed)	0
n+4	Point number			0xssss
n+6	Not used			0x0000
n+8	Axis-1 data			0xbbbbbbbb
n+10				
n+12	Axis-2 data			0xbbbbbbbb
n+14				
n+16	Axis-3 data			0xbbbbbbbb
n+18				
n+20	Axis-4 data			0xbbbbbbbb
n+22				
n+24	Axis-5 data			0xbbbbbbbb
n+26				
n+28	Axis-6 data			0xbbbbbbbb
n+30				

u :Specify the point data unit in 1 bit.

Value	Meaning
0	Pulse units
1	Millimeter units

tt :Specify in 2 bits the hand system to be defined.
Valid only when SCARA robot is specified and units are in millimeters.

Value	Meaning
01	Right-handed system is defined.
10	Left-handed system is defined.
Others	No hand system is defined.

ssss :Specify the point number in 16 bits.
Specified range: 1 (=0x0001) to 9999 (=0x270F)

bbbbbbbb :Specify the point data in 32 bits. (little endian)
Data should be integers when units are in pulses.
Data should be integers (x100) when units are in millimeters.

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the point data definition command as shown at right, to create the following point data in pulse units.

Point number = 100

Axis 1 = 10000

Axis 2 = -20000

Axis 3 = 5000

Axis 4 = -18000

Other axes = 0

Channel	Value
n	0x0100
n+2	0x0000
n+4	0x0064
n+6	0x0000
n+8	0x2710
n+10	0x0000
n+12	0xB1E0
n+14	0xFFFF
n+16	0x1388
n+18	0x0000
n+20	0xB9B0
n+22	0xFFFF
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0x0000
m+10	0x0000
m+12	0x0000
m+14	0x0000
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.3.1.2 Point data reference

Use this command to find and obtain point data by specifying the point number.

■ Command

Channel	Contents	Value
n	Command code	0x0101
n+2	Not used	0x0000
n+4	Point number	0xssss
n+6	Not used	
to		
n+30		

ssss :Specify the point number in 16 bits.
Specified range: 0 (=0x0000) to 9999 (=0x270F)

■ Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4	Point number			0xssss
m+6	Point flag	bit 0	Point unit	u
		bit 2–bit 1	Hand system	tt
		bit 15–bit 3	(0:Fixed)	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

ssss :Shows the point number in 16 bits.
Specified range: 0 (=0x0000) to 9999 (=0x270F)

u :Shows the point data unit in 1 bit.

Value	Meaning
0	Pulse units
1	Millimeter units

tt :Shows in 2 bits the hand system to define point data.
Valid only when SCARA robot is specified and units are in millimeters.

Value	Meaning
00	No hand system is defined.
01	Right-handed system is defined.
10	Left-handed system is defined.

bbbbbbb :Shows the point data in 32 bits. (little endian)
Data is shown in integers when units are in pulses.
Data is shown in integers (x100) when units are in millimeters.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the point data reference command as shown at right, to search and obtain point data at point number 50.

Channel	Value
n	0x0101
n+2	0x0000
n+4	0x0032
n+6	0x0000
n+8	0x0000
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly to obtain the following point data.

Point number = 50

Axis 1 = 100.00

Axis 2 = -200.00

Axis 3 = 50.00

Axis 4 = -180.00

Other axes = 0.00

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0032
m+6	0x0001
m+8	0x2710
m+10	0x0000
m+12	0xB1E0
m+14	0xFFFF
m+16	0x1388
m+18	0x0000
m+20	0xB9B0
m+22	0xFFFF
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.3.2 Point comment-related command

Execute this command to define or obtain point comment data.

4.3.2.1 Point comment data definition

Use this command to define point comment data by specifying the point number and point comment data.

■ Command

Channel	Contents	Value
n	Command code	0x0104
n+2	Not used	0x0000
n+4	Point number	0xssss
n+6	Not used	0x0000
n+8	Comment data	0xbbbb
n+10		0xbbbb
n+12		0xbbbb
n+14		0xbbbb
n+16		0xbbbb
n+18		0xbbbb
n+20		0xbbbb
n+22		0x00bb
n+24	Not used	0x0000
to		
n+30		

ssss :Specify the point number in 16 bits.
Specified range: 0 (=0x0000) to 9999 (=0x270F)

bb :Specify 1 byte comment data in 8 bits. (little endian)
Specified range: " "(=0x20) to "~ "(=0x7E)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the point comment data definition command as shown at right, to create the following point comment data.

Point number = 100

Comment data = "WAIT ORG"

(character code : "W" =0x57
"A" =0x41
" I " =0x49
"T" =0x54
" " =0x20
"O" =0x4F
"R" =0x52
"G" =0x47)

Channel	Value
n	0x0104
n+2	0x0000
n+4	0x0064
n+6	0x0000
n+8	0x4157
n+10	0x5449
n+12	0x4F20
n+14	0x4752
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0x0000
m+10	0x0000
m+12	0x0000
m+14	0x0000
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.3.2.2 Point comment data reference

Use this command to search and obtain point comment data by specifying the point number.

■ Command

Channel	Contents	Value
n	Command code	0x0105
n+2	Not used	0x0000
n+4	Point number	0xssss
n+6	Not used	0x0000
to		
n+30		

ssss :Specify the point number in 16 bits.
Specified range: 0 (=0x0000) to 9999 (=0x270F)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
m+4	Point number	0xssss
m+6	Not used	
m+8	Comment data	0xbbbb
m+10		0xbbbb
m+12		0xbbbb
m+14		0xbbbb
m+16		0xbbbb
m+18		0xbbbb
m+20		0xbbbb
m+22		0x00bb
m+24	Not used	
to		
m+30		

ssss :Shows the point number in 16 bits.
Specified range: 0 (=0x0000) to 9999 (=0x270F)

bb :Shows the 1 byte comment data in 8 bits. (little endian)

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the point comment data reference command as shown at right, to obtain point comment data at point number 50.

Channel	Value
n	0x0105
n+2	0x0000
n+4	0x0032
n+6	0x0000
n+8	0x0000
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly to obtain the following point data.

Point number = 50

Comment data = "WAIT ORG"

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0032
m+6	0x0000
m+8	0x4157
m+10	0x5449
m+12	0x4F20
m+14	0x4752
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.3.3 Pallet-related command

Execute this command to define or obtain pallet data.

4.3.3.1 Pallet data definition

Use this command to define pallet data by specifying the pallet number and the number of pallets (Nx, Ny, Nz).

NOTE

Point data used for pallet movement is determined by the pallet number. Refer to the robot controller user's manual or robot programming manual for detailed information.

■ Command

Channel	Contents	Value
n	Command code	0x0108
n+2	Not used	0x0000
n+4	Pallet number	0xssss
n+6	Number of pallets in X direction (Nx)	0xaaaa
n+8	Number of pallets in Y direction (Ny)	0xaaaa
n+10	Number of pallets in Z direction (Nz)	0xaaaa
n+12	Not used	0x0000
to		
n+30		

ssss :Specify the pallet number in 16 bits.
Specified range: 0 (=0x0000) to 19 (=0x0013)

aaaa :Specify the number of pallets (positive integer) in 16 bits.
Specified range: 0 (=0x0000) to 32767 (=0x7FFF)
The value of "Nx*Ny*Nz" should be 32767 or less.

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the pallet data definition command as shown at right, to create the following pallet.

Pallet number = 10
Nx = 10
Ny = 15
Nz = 1

Channel	Value
n	0x0108
n+2	0x0000
n+4	0x000A
n+6	0x000A
n+8	0x000F
n+10	0x0001
n+12	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
to	
m+30	

4.3.3.2 Pallet data reference

Use this command to obtain pallet data by specifying the pallet number.

■ Command

Channel	Contents	Value
n	Command code	0x0109
n+2	Not used	0x0000
n+4	Pallet number	0xssss
n+6	Not used	0x0000
to		
n+30		

ssss :Specify the pallet number in 16 bits.
Specified range: 0 (=0x0000) to 19 (=0x0013)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
m+4	Pallet number	0xssss
m+6	Number of pallets in X direction (Nx)	0xaaaa
m+8	Number of pallets in Y direction (Ny)	0xaaaa
m+10	Number of pallets in Z direction (Nz)	0xaaaa
m+12	Not used	
to		
m+30		

ssss :Shows the pallet number in 16 bits.
aaaa :Shows the number of pallets in 16 bits.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the pallet data reference command as shown at right, to obtain pallet data at pallet number 10.

Channel	Value
n	0x0109
n+2	0x0000
n+4	0x000A
n+6	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly to obtain the following pallet data.

Pallet number = 10

Nx = 10

Ny = 15

Nz = 1

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x000A
m+6	0x000A
m+8	0x000F
m+10	0x0001
m+12	0x0000
to	
m+30	

4.3.4 Shift-related command

Execute this command to define or obtain shift data.

4.3.4.1 Shift data definition

Use this command to define shift data by specifying the shift number and shift data.

■ Command

Channel	Contents	Value
n	Command code	0x010C
n+2	Not used	0x0000
n+4	Shift number	0xssss
n+6	Not used	0x0000
n+8	Axis-1 data	0xbbbbbbbb
n+10		
n+12	Axis-2 data	0xbbbbbbbb
n+14		
n+16	Axis-3 data	0xbbbbbbbb
n+18		
n+20	Axis-4 data	0xbbbbbbbb
n+22		
n+24	Not used	0x0000
to		
n+30		

ssss :Specify the shift number in 16 bits.
Specified range: 0 (=0x0000) to 9 (=0x0009)

bbbbbbbb :Specify the shift data in 32 bits. (little endian)
Data should by integers (x100).

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the shift data definition command as shown at right, to create the following shift data.

Shift number = 5

Axis 1 = 100.00

Axis 2 = -200.00

Axis 3 = 50.00

Axis 4 = -180.00

Channel	Value
n	0x010C
n+2	0x0000
n+4	0x0005
n+6	0x0000
n+8	0x2710
n+10	0x0000
n+12	0xB1E0
n+14	0xFFFF
n+16	0x1388
n+18	0x0000
n+20	0xB9B0
n+22	0xFFFF
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0x0000
m+10	0x0000
m+12	0x0000
m+14	0x0000
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.3.4.2 Shift data reference

Use this command to search and obtain shift data by specifying the shift number.

■ Command

Channel	Contents	Value
n	Command code	0x010D
n+2	Not used	0x0000
n+4	Shift number	0xssss
n+6	Not used	0x0000
to		
n+30		

ssss :Specify the shift number in 16 bits.
Specified range: 0 (=0x0000) to 9 (=0x0009)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
m+4	Shift number	0xssss
m+6	Not used	
m+8	Data 1	0xbbbbbbbb
m+10		
m+12	Data 2	0xbbbbbbbb
m+14		
m+16	Data 3	0xbbbbbbbb
m+18		
m+20	Data 4	0xbbbbbbbb
m+22		
m+24	Not used	
to		
m+30		

ssss :Shows the shift number in 16 bits.
bbbbbbbb :Shows the shift data in 32 bits. (little endian)
Data is show in integers (x100).

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.
ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the shift data reference command as shown at right, to obtain shift data at shift number 5.

Channel	Value
n	0x010D
n+2	0x0000
n+4	0x0005
n+6	0x0000
n+8	0x0000
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly to obtain the following shift data.

Shift number = 5

Axis 1 = 100.00

Axis 2 = -200.00

Axis 3 = 50.00

Axis 4 = -180.00

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0005
m+6	0x0000
m+8	0x2710
m+10	0x0000
m+12	0xB1E0
m+14	0xFFFF
m+16	0x1388
m+18	0x0000
m+20	0xB9B0
m+22	0xFFFF
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.3.5 Hand-related command

Execute this command to define or obtain hand data.

4.3.5.1 Hand data definition

Use this command to define hand data by specifying the hand number and each data.

■ Command

Channel	Contents		Value
n	Command code	For main robot	0x0110
		For sub robot	0x0190
n+2	Not used		0x0000
n+4	Hand number		0xssss
n+6	Not used		0x0000
n+8	Data 1		0xbbbbbbbb
n+10			
n+12	Data 2		0xbbbbbbbb
n+14			
n+16	Data 3		0xbbbbbbbb
n+18			
n+20	Data 4		0xbbbbbbbb
n+22			
n+24	Not used		0x0000
to			
n+30			

ssss :Specify the hand number in 16 bits.
Specified range for main robot : 0 (0x0000) to 3 (=0x0003)
Specified range for sub robot : 4 (=0x0004) to 7 (=0x0007)

bbbbbbbb :When SCARA robot is specified and data 4 is 0:
Data 1 : Specify the integer in 32 bits. (little endian)
Data 2 and 3 : Specify the integer (x100) in 32 bits. (little endian)
Data 4 : When hand is installed to R-axis =1, other cases =0
In other cases
Data 1 to 3 : Specify the integer (x100) in 32 bits. (little endian)
Data 4 : When hand is installed to R-axis =1, other cases =0

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.
ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the hand data definition command as shown at right, to create hand data for a Cartesian robot.

Hand number = 1
Data 1 = 100.00
Data 2 = -20.00
Data 3 = 50.00
Data 4 = 0

Channel	Value
n	0x0110
n+2	0x0000
n+4	0x0001
n+6	0x0000
n+8	0x2710
n+10	0x0000
n+12	0xF830
n+14	0xFFFF
n+16	0x1388
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0x0000
m+10	0x0000
m+12	0x0000
m+14	0x0000
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.3.5.2 Hand data reference

Use this command to obtain hand data by specifying the hand number.

■ Commands

Channel	Contents	Value
n	Command code	0x0111
n+2	Not used	0x0000
n+4	Hand number	0xssss
n+6	Not used	0x0000
to		
n+30		

ssss :Specify the hand number in 16 bits.
Specified range: 0 (0x0000) to 7 (=0x0007)
(Numbers 0 to 3 are hand data for main robot, and numbers 4 to 7 are hand data for sub robot.)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
m+4	Hand number	0xssss
m+6	Not used	
m+8	Data 1	0xbbbbbbbb
m+10		
m+12	Data 2	0xbbbbbbbb
m+14		
m+16	Data 3	0xbbbbbbbb
m+18		
m+20	Data 4	0xbbbbbbbb
m+22		
m+24	Not used	
to		
m+30		

ssss :Shows the hand number in 16 bits.

bbbbbbbb :When SCARA robot is specified and data 4 is 0:
Data 1 : Shows the integer in 32 bits. (little endian)
Data 2 and 3 : Shows the integer (x100) in 32 bits. (little endian)
Data 4 : When hand is installed to R-axis =1, other cases =0
In other cases
Data 1 to 3 : Shows the integer (x100) in 32 bits. (little endian)
Data 4 : When hand is installed to R-axis =1, other cases =0

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the hand data reference command as shown at right, to obtain hand data.

Channel	Value
n	0x0111
n+2	0x0000
n+4	0x0001
n+6	0x0000
n+8	0x0000
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly to obtain the following hand data.

Hand number = 1

Data 1 = 10000

Data 2 = -20.00

Data 3 = 50.00

Data 4 = 0

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0001
m+6	0x0000
m+8	0x2710
m+10	0x0000
m+12	0xF830
m+14	0xFFFF
m+16	0x1388
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.4 Category 3 remote commands

Category 3 remote commands are arithmetic commands. A command list is given below.

No.	Command contents			Command code n
1	Static variable-related commands	Assignment	Value	0x0200
			Variable	0x0201
		Addition	Value	0x0204
			Variable	0x0205
		Subtraction	Value	0x0208
			Variable	0x0209
		Multiplication	Value	0x020C
			Variable	0x020D
		Division	Value	0x0210
			Variable	0x0211
		Reference	Variable	0x0214
2	Parameter-related command	Assignment	Main robot	0x0220
			Sub robot	0x02A0
		Reference	Main robot	0x0224
			Sub robot	0x02A4
3	Point-related command	Point assignment		0x0230
		Addition		0x0234
		Subtraction		0x0235
		Pallet point assignment		0x0238
4	Element assignment command	Point element assignment	"x1" input format	0x0240
			"x100" input format	0x0241
		Shift element assignment	"x100" input format	0x0245

m: Start address of the input area assigned to the master module
n : Start address of the output area assigned to the master module
("m" indicates the data direction from remote to master, and "n" the data direction from master to remote.)

4.4.1 Static variable-related command

Execute this command to assign a numerical value to a static variable for four arithmetic operations or reference.

4.4.1.1 Assigning a numerical value to a static variable

This command assigns a numerical value to a static variable (SGIn or SGRn) by specifying the destination variable number and the numerical value.

Variable number 1 = numerical value

CAUTION

- **A real number is assigned when a real variable was used.**
- **Due to cancellation of significant digits when using real number data for assignment reference, the assigned data might sometimes differ from the reference data.**

■ Command

Channel	Contents	Value
n	Command code	0x0200
n+2	Not used	0x0000
n+4	Variable number 1 (Variable number at assignment destination)	0xssss
n+6	Not used	0x0000
n+8	Numerical data	0xbbbbbbbb
n+10		
n+12	Not used	0x0000
to		
n+30		

ssss :Specify variable number 1 in 16 bits.
Specified range for integer variable: 0 (0x0000) to 7 (=0x0007)
Specified range for real variable : 256 (=0x0100) to 263 (=0x0107)

Integer variable	Variable number	Real variable	Variable number
SGI0	0(=0x0000)	SGR0	256(=0x0100)
SGI1	1(=0x0001)	SGR1	257(=0x0101)
:	:	:	:
SGI7	7(=0x0007)	SGR7	263(=0x0107)

bbbbbbbb :Specify the integer in 32 bits. (little endian)
Specify a signed integer value when assigning to an integer variable.
Specify a single-precision real number when assigning to a real variable.

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.
ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to assign numerical data to variable number 1.

Variable number 1 = 1

Numerical data = 10000

Channel	Value
n	0x0200
n+2	0x0000
n+4	0x0001
n+6	0x0000
n+8	0x2710
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0x0000
m+10	0x0000
m+12	0x0000
m+14	0x0000
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.4.1.2 Assigning a variable to a static variable

This command assigns a numerical value to a static variable (SGIn or SGRn) by designating the source variable number and destination variable number.

Variable number 1 = Variable number 2

■ Command

Channel	Contents	Value
n	Command code	0x0201
n+2	Not used	0x0000
n+4	Variable number 1 (Variable number at assignment destination)	0xssss
n+6	Not used	0x0000
n+8	Variable number 2 (Variable number at assignment source)	0xssss
n+10	Not used	0x0000
to		
n+30		

ssss :Specify variable numbers 1 and 2 in 16 bits.
Specified range for integer variable: 0 (0x0000) to 7 (=0x0007)
Specified range for real variable : 256 (=0x0100) to 263 (=0x0107)

Integer variable	Variable number	Real variable	Variable number
SGI0	0(=0x0000)	SGR0	256(=0x0100)
SGI1	1(=0x0001)	SGR1	257(=0x0101)
:	:	:	:
SGI7	7(=0x0007)	SGR7	263(=0x0107)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to assign numerical data of variable number 2 to variable number 1.

Variable number 1 = 1

Variable number 2 = 2

Channel	Value
n	0x0201
n+2	0x0000
n+4	0x0001
n+6	0x0000
n+8	0x0002
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0x0000
m+10	0x0000
m+12	0x0000
m+14	0x0000
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.4.1.3 Arithmetic operation using numerical data on static variable

This command performs four arithmetic operations by specifying variable number 1 and a numerical value. Results are stored in a static variable (SGIn or SGRn) specified by variable number 1.

Variable number 1 = Variable number 1 (operator) numerical value

■ Command

Channel	Contents		Value
n	Command code	Addition	0x0204
		Subtraction	0x0208
		Multiplication	0x020C
		Division	0x0210
n+2	Not used		0x0000
n+4	Variable number 1 (Variable number at addition destination)		0xssss
n+6	Not used		0x0000
n+8	Numerical data		0xbbbbbbbb
n+10			
n+12	Not used		0x0000
to			
n+30			

ssss :Specify variable number 1 in 16 bits.
Specified range for integer variable: 0 (0x0000) to 7 (=0x0007)
Specified range for real variable : 256 (=0x0100) to 263 (=0x0107)

Integer variable	Variable number	Real variable	Variable number
SGI0	0(=0x0000)	SGR0	256(=0x0100)
SGI1	1(=0x0001)	SGR1	257(=0x0101)
:	:	:	:
SGI7	7(=0x0007)	SGR7	263(=0x0107)

bbbbbbbb :Specify the integer in 32 bits. (little endian)
Specify a signed integer value when assigning to an integer variable.
Specify a single-precision real number when assigning to a real variable.

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command to assign numerical data to a static variable as shown at right.

Variable number 1 = 1

Numerical data = 10000

Channel	Value
n	0x0204
n+2	0x0000
n+4	0x0001
n+6	0x0000
n+8	0x2710
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0x0000
m+10	0x0000
m+12	0x0000
m+14	0x0000
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.4.1.4 Arithmetic operation using variable on static variable

This command performs four arithmetic operations by specifying variable numbers 1 and 2. Results are stored in a static variable (SGIn or SGRn) specified by variable number 1.

Variable number 1 = Variable number 1 (operator) variable number 2

■ Command

Channel	Contents		Value
n	Command code	Addition	0x0205
		Subtraction	0x0209
		Multiplication	0x020D
		Division	0x0211
n+2	Not used		0x0000
n+4	Variable number 1 (Variable number at arithmetic operation destination)		0xssss
n+6	Not used		0x0000
n+8	Variable number 2 (Variable number at arithmetic operation source)		0xssss
n+10	Not used		0x0000
to			
n+30			

ssss :Specify variable numbers 1 and 2 in 16 bits.
Specified range for integer variable: 0 (0x0000) to 7 (=0x0007)
Specified range for real variable : 256 (=0x0100) to 263 (=0x0107)

Integer variable	Variable number	Real variable	Variable number
SGI0	0(=0x0000)	SGR0	256(=0x0100)
SGI1	1(=0x0001)	SGR1	257(=0x0101)
:	:	:	:
SGI7	7(=0x0007)	SGR7	263(=0x0107)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this arithmetic operation command to multiply static variables as shown at right.

Variable number 1 = 1

Variable number 2 = 2

Channel	Value
n	0x020D
n+2	0x0000
n+4	0x0001
n+6	0x0000
n+8	0x0002
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0x0000
m+10	0x0000
m+12	0x0000
m+14	0x0000
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.4.1.5 Static variable value reference

Use this command to search and obtain the value stored in a static variable (SGIn or SGRn) by specifying the variable number.

■ Command

Channel	Contents	Value
n	Command code	0x0214
n+2	Not used	0x0000
n+4	Variable number	0xssss
n+6	Not used	0x0000
to		
n+30		

ssss :Specify variable number in 16 bits.
Specified range for integer variable: 0 (0x0000) to 7 (=0x0007)
Specified range for real variable : 256 (=0x0100) to 263 (=0x0107)

Integer variable	Variable number	Real variable	Variable number
SGI0	0(=0x0000)	SGR0	256(=0x0100)
SGI1	1(=0x0001)	SGR1	257(=0x0101)
:	:	:	:
SGI7	7(=0x0007)	SGR7	263(=0x0107)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
m+4	Variable number	0xssss
m+6	Not used	
m+8	Value of variable	0xbbbbbbbb
m+10		
m+12	Not used	
to		
m+30		

ssss :Specify variable number in 16 bits.
Specified range for integer variable: 0 (0x0000) to 7 (=0x0007)
Specified range for real variable : 256 (=0x0100) to 263 (=0x0107)

bbbbbbbb :Shows the numerical value in 32 bits. (little endian)
Specify a signed integer value when assigning to an integer variable.
Specify a single-precision real number when assigning to a real variable.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to obtain the numerical value of variable number 5.

Channel	Value
n	0x0214
n+2	0x0000
n+4	0x0005
n+6	0x0000
n+8	0x0000
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly to obtain the following variable.

Variable number = 5

Value = 50

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0005
m+6	0x0000
m+8	0x0032
m+10	0x0000
m+12	0x0000
m+14	0x0000
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.4.2 Parameter-related command

Execute this command to assign a value to a parameter or obtain a parameter.

4.4.2.1 Assigning a value to a parameter

This command assigns a numerical value to a specified parameter by specifying the parameter number, axis and numerical value.

Robot parameter		Parameter number	Assignment range
WEIGHT	Robot payload (kg)	1(=0x0001)	0 to maximum payload

Axis parameter		Parameter number	Assignment range
ACCEL	Acceleration coefficient	257(=0x0101)	1 to 100
DECEL	Deceleration ratio	258(=0x0102)	1 to 100
TOLE	Tolerance (pulses)	259(=0x0103)	1 to 2048
OUTPOS	OUT effective position (pulses)	260(=0x0104)	1 to 614400
ARCH	Arch position (pulses)	261(=0x0105)	1 to 614400
AXWGHT	Axis payload (kg)	262(=0x0106)	0 to maximum payload
TORQUE	Torque (%)	263(=0x0107)	1 to 100

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0220
			For sub robot	0x02A0
n+2	Not used			0x0000
n+4	Parameter number			0xssss
n+6	Specified axis	bit 0	Axis 1	0x00tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 15–bit 6	(0:Fixed)	
n+8	Numerical data			0xbbbbbbbb
n+10				
n+12	Not used			0x0000
to				
n+30				

ssss :Specify the parameter number in 16 bits.

tt :Specify the axis number in bit pattern using lower 8 bits.
Only one axis can be specified.
Specify "0" for robot parameters.

bbbbbbbb :Specify the integer in 32 bits. (little endian)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to assign a numerical value to the tolerance for axis 3 of the main robot.

Parameter number = 259

Specified axis = 3

Numerical data = 1000

Channel	Value
n	0x0220
n+2	0x0000
n+4	0x0103
n+6	0x0004
n+8	0x03E8
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0x0000
m+10	0x0000
m+12	0x0000
m+14	0x0000
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.4.2.2 Parameter value reference

Use this command to search and obtain parameter setting data by specifying the parameter number.

Robot parameter		Parameter number	Reference range
WEIGHT	Robot payload (kg)	1(=0x0001)	0 to maximum payload

Axis parameter		Parameter number	Reference range
ACCEL	Acceleration coefficient	257(=0x0101)	1 to 100
DECEL	Deceleration ratio	258(=0x0102)	1 to 100
TOLE	Tolerance (pulses)	259(=0x0103)	1 to 2048
OUTPOS	OUT effective position (pulses)	260(=0x0104)	1 to 614400
ARCH	Arch position (pulses)	261(=0x0105)	1 to 614400
AXWGHT	Axis payload (kg)	262(=0x0106)	0 to maximum payload

■ Command

Channel	Contents			Value
n	Command code		For main robot	0x0224
			For sub robot	0x02A4
n+2	Not used			0x0000
n+4	Parameter number			0xssss
n+6	Specified axis	bit 0	Axis 1	0x00tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 15–bit 6	(0:Fixed)	
n+8	Not used			0x0000
to				
n+30				

ssss :Specify the parameter number in 16 bits.

tt :Specify the axis number in bit pattern using lower 8 bits.
Only one axis can be specified.
Specify "0" for robot parameters.

■ Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4	Parameter number			0xssss
m+6	Specified axis	bit 0	Axis 1	0x00tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 15–bit 6	Not used	
m+8	Numerical data			0xbbbbbbbb
m+10				
m+12	Not used			
to				
m+30				

ssss :Specify the parameter number in 16 bits.

tt :Specify the axis number in bit pattern using lower 8 bits.
Only one axis can be specified.
Specify "0" for robot parameters.

bbbbbbbb :Specify the integer in 32 bits. (little endian)

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to obtain the OUT effective position of axis 1 of the main robot.

Parameter number = 260

Specified axis = 1

Channel	Value
n	0x0224
n+2	0x0000
n+4	0x0104
n+6	0x0001
n+8	0x0000
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly to obtain the following parameter.

Parameter number = 260

Specified axis = 1

Numerical data = 131071

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0104
m+6	0x0001
m+8	0xFFFF
m+10	0x0001
m+12	0x0000
m+14	0x0000
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.4.3 Point-related command

Execute this command to assign a point to a parameter or obtain a parameter.

4.4.3.1 Assigning a point to a parameter

This command assigns a numerical value to a specified parameter by specifying the parameter number, axis and numerical value.

Point number 1 = Point number 2

■ Command

Channel	Contents	Value
n	Command code	0x0230
n+2	Not used	0x0000
n+4	Point number 1 (Point number at assignment destination)	0xssss
n+6	Point number 2 (Point number at assignment source)	0xssss
n+8	Not used	0x0000
to		
n+30		

ssss :Specify the point number in 16 bits.
Specified range: 0 (= 0x0000) to 9999 (=0x270F)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to assign a point to the specified point.

Point number 1 = 1

Point number 2 = 100

Channel	Value
n	0x0230
n+2	0x0000
n+4	0x0001
n+6	0x0064
n+8	0x0000
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0x0000
m+10	0x0000
m+12	0x0000
m+14	0x0000
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.4.3.2 Point addition/subtraction

This command adds and subtracts points by specifying point number 1 and point number 2.
Point number 1 = Point number 1 (operator) point number 2

■ Command

Channel	Contents		Value
n	Command code	Addition	0x0234
		Subtraction	0x0235
n+2	Not used		0x0000
n+4	Point number 1 (Point number at operation destination)		0xssss
n+6	Point number 2 (Point number at operation source)		0xssss
n+8	Not used		0x0000
to			
n+30			

ssss :Specify the point number in 16 bits.
Specified range: 0 (= 0x0000) to 9999 (=0x270F)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.
ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the point addition command as shown at right, to add point number 2 to point number 1.

Point number 1 = 1

Point number 2 = 100

Channel	Value
n	0x0234
n+2	0x0000
n+4	0x0001
n+6	0x0064
n+8	0x0000
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0x0000
m+10	0x0000
m+12	0x0000
m+14	0x0000
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.4.3.3 Assigning a pallet point

This command assigns a pallet point to the destination point number by specifying a pallet number and work position number.

Pallet point number = Pallet point (pallet number, work position number)

NOTE

- The target pallet must be defined.
- The maximum value of work position number is determined by the target pallet definition.

■ Command

Channel	Contents	Value
n	Command code	0x0238
n+2	Not used	0x0000
n+4	Point number (Point number at assignment destination)	0xssss
n+6	Pallet number	0xaaaa
n+8	Work position number	0xbbbb
n+10	Not used	0x0000
to		
n+30		

ssss :Specify the point number in 16 bits.
Specified range: 0 (=0x0000) to 9999 (=0x270F)

aaaa :Specify the pallet number in 16 bits.
Specified range: 0 (=0x0000) to 19 (=0x0013)

bbbb :Specify the work position number in 16 bits.
Specified range: 1 (=0x0000) to 32767 (=0x7FFF)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to assign a pallet point to the following point.

Point number = 100

Pallet number = 2

Work position number = 133

Channel	Value
n	0x0238
n+2	0x0000
n+4	0x0064
n+6	0x0002
n+8	0x0085
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0x0000
m+10	0x0000
m+12	0x0000
m+14	0x0000
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.4.4 Element assignment command

Execute this command to assign a number to a point or shift element.

4.4.4.1 Assigning to a point element

This command assigns a numerical value to a point element by specifying the point number, data number and numerical value.

LOC [data number] (point number) = numerical value

NOTE

When 1000 is specified in the "x1" input format as a numerical value, 1000 is assigned.
When 1000 is specified in the "x100" input format as a numerical value, 10.00 is assigned.
Use the proper input format according to the point data format of the assignment destination.

■ Command

Channel	Contents			Value
n	Command code		"x1" input format	0x0240
			"x100" input format	0x0241
n+2	Not used			0x0000
n+4	Point number (Point number at assignment destination)			0xssss
n+6	Data number designation	bit 0	Data 1	0x00tt
		bit 1	Data 2	
		bit 2	Data 3	
		bit 3	Data 4	
		bit 4	Data 5	
		bit 5	Data 6	
		bit 15–bit 6	(0:Fixed)	
n+8	Numerical value			0xbbbbbbbb
n+10				
n+12	Not used			0x0000
to				
n+30				

ssss :Specify the point number in 16 bits.
Specified range: 0 (0x0000) to 9999 (=0x270F)

tt :Specify the data number in bit pattern using lower 6 bits.

bbbbbbbb :Specify the integer in 32 bits. (little endian)
Specify data in integers when using "x1" input format.
Specify data in integers (x100) when using "x100" input format.

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to assign a numerical value to part of the following point.

Point number = 1

Data number designation = 4

Numerical value = 10.00

Channel	Value
n	0x0241
n+2	0x0000
n+4	0x0001
n+6	0x0008
n+8	0x03E8
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0x0000
m+10	0x0000
m+12	0x0000
m+14	0x0000
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.4.4.2 Assigning to a shift element

This command assigns a numerical value to a shift element by specifying the shift number, data number and numerical value.

LOC [data number] (shift number) = numerical value

■ Command

Channel	Contents			Value
n	Command code			0x0245
n+2	Not used			0x0000
n+4	Shift number (Shift number at assignment destination)			0xssss
n+6	Data number designation	bit 0	Data 1	0x00tt
		bit 1	Data 2	
		bit 2	Data 3	
		bit 3	Data 4	
		bit 15–bit 4	(0:Fixed)	
n+8	Numerical value			0xbbbbbbbb
n+10				
n+12	Not used			0x0000
to				
n+30				

ssss :Specify the shift number in 16 bits.
Specified range: 0 (0x0000) to 9 (=0x0009)

tt :Specify the data number in bit pattern using lower 4 bits.

bbbbbbbb :Specify the integer (x100) in 32 bits. (little endian)

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to assign a real number value to part of the following shift.

Shift number = 1

Data number designation = 2

Numerical value = 10.00

Channel	Value
n	0x0245
n+2	0x0000
n+4	0x0001
n+6	0x0002
n+8	0x03E8
n+10	0x0000
n+12	0x0000
n+14	0x0000
n+16	0x0000
n+18	0x0000
n+20	0x0000
n+22	0x0000
n+24	0x0000
n+26	0x0000
n+28	0x0000
n+30	0x0000

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0x0000
m+10	0x0000
m+12	0x0000
m+14	0x0000
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.5 Category 4 remote commands

Category 4 remote commands are I/O port commands. A command list is given below.

No.	Command contents			Command code n
1	I/O port command	Assignment	port units	0x0300
		Assignment	bit units	0x0301
		Reference	port units	0x0304

m: Start address of the input area assigned to the master module
n : Start address of the output area assigned to the master module
("m" indicates the data direction from remote to master, and "n" the data direction from master to remote.)

4.5.1 I/O port commands

Use these commands to assign a value to an I/O port or obtain the contents of a specified I/O port.

4.5.1.1 Assigning a numerical value to an I/O port

This command assigns a bit pattern to a port number by specifying the destination port number and bit pattern.

■ Command

Channel	Contents			Value
n	Command code		Port units	0x0300
			Bit units	0x0301
n+2	Not used			0x0000
n+4	Port number	bit 3–bit 0	Bit number	g
		bit 7–bit 4	Units of port number	r
		bit 11–bit 8	Tens of port number	q
		bit 15–bit 12	Specified port type	p
n+6	Assignment bit pattern			0x00bb
n+8	Not used			0x0000
to				
n+30				

g :Specify the bit number in 4 bits.
Specified range: 0 to 7

r, q :Specify the place of each port number in 4 bits.

p :Specify the port type in 4 bits.
When in port units, specify 0 in the bit number.

Designated port type	Bit pattern	Specified range of port number
DO	0001	2 to 7,10 to 17,20 to 27
MO	0010	2 to 7,10 to 17,20 to 27
LO	0011	0
TO	0100	0
SO	0110	2 to 7,10 to 17,20 to 27

bb :Specify the bit pattern in 8 bits.
When in bit units, use 0 or 1 to specify the bit pattern.

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to output a numerical value to the following output port.

Output port = DO12 ()

Numerical data = 7

Channel	Value
n	0x0300
n+2	0x0000
n+4	0x1120
n+6	0x0007
n+8	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
to	
m+30	

Example:

Use this command as shown at right, to output a numerical value to the following output port.

Output port = DO (21)

Numerical data = 1

Channel	Value
n	0x0301
n+2	0x0000
n+4	0x1021
n+6	0x0001
n+8	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
to	
m+30	

4.5.1.2 I/O port reference

Use this command to obtain the contents of a port number by specifying the port number.

■ Command

Channel	Contents			Value
n	Command code		Port units	0x0304
n+2	Not used			0x0000
n+4	Port number	bit 3–bit 0	(0:Fixed)	0
		bit 7–bit 4	Units of port number	r
		bit 11–bit 8	Tens of port number	q
		bit 15–bit 12	Specified port type	p
n+6	Not used			0x0000
to				
n+30				

r, q :Specify the place of each port number in 4 bits.

p :Specify the port type in 4 bits.

Designated port type	Bit pattern	Specified range of port number
DI	0000	0 to 7,10 to 17,20 to 27
DO	0001	0 to 7,10 to 17,20 to 27
MO	0010	0 to 7,10 to 17,20 to 27
LO	0011	0
TO	0100	0
SI	0101	0 to 7,10 to 17,20 to 27
SO	0110	0 to 7,10 to 17,20 to 27

■ Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4	Port number	bit 3–bit 0	Not used	0
		bit 7–bit 4	Units of port number	r
		bit 11–bit 8	Tens of port number	q
		bit 15–bit 12	Specified port type	p
m+6	Bit pattern			0x00bb
m+8	Not used			
to				
m+30				

r, q :Shows the place of each port number in 4 bits.

p :Shows the port type in 4 bits.

bb :Shows the bit pattern in 8 bits.
When in bit units, 0 or 1 is used to show the bit pattern.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to obtain the following port data.

Output port = DO12 ()

Channel	Value
n	0x0304
n+2	0x0000
n+4	0x1120
n+6	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Output port = DO12 ()

Numerical data = 7

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x1120
m+6	0x0007
m+8	0x0000
to	
m+30	

Example:

Use this command as shown at right, to output a numerical value to the following port data.

Input port = DI2 ()

Channel	Value
n	0x0304
n+2	0x0000
n+4	0x0020
n+6	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Input port = DI2 ()

Numerical data = 127

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0020
m+6	0x007F
m+8	0x0000
to	
m+30	

4.6 Category 5 remote commands

Category 5 remote commands are key operation commands. A command list is given below.

NOTE

Category 5 commands can be used in AUTO mode.

No.	Command contents		Command code n
1	Execution program designation		0x0401
2	Program execution	Program execution	0x0402
		Program step execution	0x0403
		Program skip execution	0x0404
		Program next execution	0x0405
3	Program reset		0x0406
4	Program task switching		0x0407
5	Program execution information reference		0x0408

m: Start address of the input area assigned to the master module
n : Start address of the output area assigned to the master module
("m" indicates the data direction from remote to master, and "n" the data direction from master to remote.)

* Key operation commands are the same as key instructions from the programming box. Normal status signifies that key input was received correctly and does not mean the actual operation was executed.
* Upon receiving a normal status after issuing a key command, allow a time interval of at least 100ms before issuing the next command. This interval will prevent errors that may occur when issuing consecutive commands.
* Check the robot program in-progress status output signal (SO13) to verify a program execution command has been run.
* Check the program reset status output signal (SO14) to verify the program reset command has been run.

4.6.1 Execution program designation

Use this command to execute a robot program.

■ Command

Channel	Contents	Value
n	Command code	0x0401
n+2	Not used	0x0000
n+4	Program name	0xbbbb
n+6		0xbbbb
n+8		0xbbbb
n+10		0xbbbb
n+12	Not used	0x0000
to		
n+30		

bb :Specify the 1-byte program name in 8 bits. (little endian)
Specify a program name with letters (uppercase), numbers and underscores (_).
When the program name is shorter than 8 characters, use a space.

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to specify a program name "ABC_DE".

Channel	Value
n	0x0401
n+2	0x0000
n+4	0x4241
n+6	0x5F43
n+8	0x4544
n+10	0x2020
n+12	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
to	
m+30	

4.6.2 Program execution

These commands execute robot program operations. These are only valid in AUTO mode.

Command	Meaning
Program execution	Starts automatic operation of a robot program. Performs the same processing as the RUN key on RPB and start input (SI12). Use the program in-progress status output signal (SO13) to verify the program is in progress.
Program step execution	Executes one line in the robot program. Enters the subroutine when a GOSUB statement is used. Performs the same processing as the STEP key (F11) on RPB.
Program skip execution	Skips one line in the program. Performs the same processing as the SKIP key (F12) on RPB.
Program next execution	Executes one line in the robot program. Executes the entire subroutine when a GOSUB statement is used. Performs the same processing as the NEXT key (F13) on RPB.

■ Command

Channel	Contents		Value
n	Command code	Program execution	0x0402
		Program step execution	0x0403
		Program skip execution	0x0404
		Program next execution	0x0405
n+2	Not used		0x0000
to			
n+30			

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use these commands to execute a program as shown at right.

Channel	Value
n	0x0402
n+2	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
to	
m+30	

4.6.3 Program reset

This command resets the robot program. This is only valid in AUTO mode. Check the program reset status output signal (SO14) to verify the program has been reset.

■ Command

Channel	Contents	Value
n	Command code	0x0406
n+2	Not used	0x0000
to		
n+30		

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command to reset a program as shown at right.

Channel	Value
n	0x0406
n+2	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
to	
m+30	

4.6.4 Program task switching

This command switches tasks to run, when the robot program is stopped. This is only valid in AUTO mode.

■ Command

Channel	Contents	Value
n	Command code	0x0407
n+2	Not used	0x0000
to		
n+30		

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
to		
m+30		

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command to switch tasks as shown at right.

Channel	Value
n	0x0407
n+2	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
to	
m+30	

4.6.5 Program execution information reference

Execute this command to acquire information on task execution, when the robot program is stopped. This is only valid in AUTO mode.

■ Command

Channel	Contents	Value
n	Command code	0x0408
n+2	Not used	0x0000
to		
n+30		

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
m+4	Program name	0xbbbb
m+6		0xbbbb
m+8		0xbbbb
m+10		0xbbbb
m+12	Task number	0xtttt
m+14	Execution line number	0xllll
m+16	Task priority	0xpppp
m+18	Not used	
to		
m+30		

bb :Shows the 1-byte program name in 8 bits. (little endian).
Program names are shown with letters (uppercase), numbers and underscores (_).
A space indicates a portion in the program name not having all 8 characters.

tttt :Shows the urrently selected task number (1 to 8).

llll :Shows the currently executed line of selected task (1 to 9999). A value + 10000 is shown when COMMON program is running.

pppp :Shows the priority of curretly selected task (17 to 47).

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number i upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command to switch program tasks as shown at right.

Channel	Value
n	0x0408
n+2	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly to switch to the following program task.

Program name = "ABCDEFGH"

Task number = 2

Execution number = 101

Task priority = 32

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x4241
m+6	0x4443
m+8	0x4645
m+10	0x4847
m+12	0x0002
m+14	0x0065
m+16	0x0020
m+18	0x0000
to	
m+30	

4.7 Category 6 remote commands

Category 6 remote commands are data handling commands. A command list is given below.

No.	Command contents			Command code n
1	Version information reference			0x0501
2	Controller configuration reference			0x0502
3	Servo status reference			0x0503
4	Absolute reset status reference			0x0504
5	Current position reference	Pulse units	For main robot	0x0505
			For sub robot	0x0585
		Millimeter units	For main robot	0x0506
			For sub robot	0x0586
6	Task status reference			0x0507
7	Task execution reference			0x0508
8	Message reference			0x0509
9	Speed status reference			0x050A
10	Arm designation status reference			0x050B
11	Arch arm status reference			0x050C
12	Service mode status reference			0x050D
13	Point unit status reference			0x050E
14	Return-to-origin status reference			0x050F

m: Start address of the input area assigned to the master module
n : Start address of the output area assigned to the master module
("m" indicates the data direction from remote to master, and "n" the data direction from master to remote.)

4.7.1 Version information reference

Execute this command to acquire information on the sotware version used in the robot controller.

■ Command

Channel	Contents	Value
n	Command code	0x0501
n+2	Not used	0x0000
to		
n+30		

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
m+4	Host software version	0xaabb
m+6	Host software revision	0xcccc
m+8	Axis-1 driver software version	0xddee
m+10	Axis-2 driver software version	0xddee
m+12	Axis-3 driver software version	0xddee
m+14	Axis-4 driver software version	0xddee
m+16	Axis-5 driver software version	0xddee
m+18	Axis-6 driver software version	0xddee
m+20	Axis-7 driver software version	0xddee
m+22	Axis-8 driver software version	0xddee
m+24	Not used	
to		
m+30		

aabb :Shows the controller's host software version in upper 8 bits and lower 8 bits.

cccc :Shows the controller's host softwar revision in 16 bits.

ddee :Shows the controller's driver software version in uper 8 bits and lower 8 bits.
For axes that do not exist actually, the value is 0x0FFF.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the categorynumber in lower 8 bis.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bis.

Example:

Use this command to obtain a software version as shown at right.

Channel	Value
n	0x0501
n+2	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Host software version : V8.08
Host software revision : R1013
Axis-1 driver software version : V1.01
Axis-2 driver software version : V1.01
Axis-3 driver softwre version : V1.01
Axis-4 driver software version : V1.01

No other axis exists.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0808
m+6	0x03F5
m+8	0x0101
m+10	0x0101
m+12	0x0101
m+14	0x0101
m+16	0x0FFF
m+18	0x0FFF
m+20	0x0FFF
m+22	0x0FFF
m+24	0x0000
to	
m+30	

4.7.2 Controller configuration reference

Execute this command to acquire information on the settings made for the robot controller.

■ Command

Channel	Contents	Value
n	Command code	0x0502
n+2	Not used	0x0000
to		
n+30		

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
m+4	Main robot number	0xaaaa
m+6	Sub robot number	0xaaaa
m+8	Axis-1 robot number	0xaaaa
m+10	Axis-2 robot number	0xaaaa
m+12	Axis-3 robot number	0xaaaa
m+14	Axis-4 robot number	0xaaaa
m+16	Axis-5 robot number	0xaaaa
m+18	Axis-6 robot number	0xaaaa
m+20	Axis-7 robot number	0xaaaa
m+22	Axis-8 robot number	0xaaaa
m+24	Unit number of option slot No. 1	0xpppp
m+26	Unit number of option slot No. 2	0xpppp
m+28	Unit number of option slot No. 3	0xpppp
m+30	Unit number of option slot No. 4	0xpppp

aaaa :Showsthe robot number.
Th robot number is determined before shipment acording to the uer specifications.

pppp :Shows the option slot unit No.

No.	Unit
0	Non
6	DIO unit (NPN specifications)
7	DIO unit (PNP specifications)
16	CC-Link unit
17	DeviceNet unit
18	Profibus unit
19	Ethernet unit
21	YC-Link unit
27	EtherNet/IP unit

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional infrmation section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use thi command to obtain the robot configuration as shown at right.

Channel	Value
n	0x0502
n+2	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Main robot number : 2000 (YK250X)
Sub robot number : 100 (no robot)
Axis-1 robot number : 2000 (YK250X)
Axis2 robot number : 200 (YK250X)
Axis-3 robot number : 200 (YK250X)
Axs-4 robot number : 2000 (YK250X)
Axis-5 robot number : 0 (no axis)
Axis-6 robot number : 0 (no axis)
Axis-7 robot number : 0 (no axis)
Axis-8 robot number : 0 (no axis)
Option slot 1 : 27 (EtherNet/IP unit)
Option slot 2 : 6 (DIO unit (NPN specifications))
Option slot 3 : 6 (DIO unit (NPN specifications))
Option slot 4 : 0 (no unit)

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x07D0
m+6	0x0064
m+8	0x07D0
m+10	0x07D0
m+12	0x07D0
m+14	0x07D0
m+16	0x0000
m+18	0x0000
m+20	0x0000
m+22	0x0000
m+24	0x001B
m+26	0x0006
m+28	0x0006
m+30	0x0000

4.7.3 Servo status reference

Execute this command to acquire information on servo status.

■ Command

Channel	Contents	Value
n	Command code	0x0503
n+2	Not used	0x0000
to		
n+30		

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
m+4	Axis-1 information	0xaaaa
m+6	Axis-2 information	0xaaaa
m+8	Axis-3 information	0xaaaa
m+10	Axis-4 information	0xaaaa
m+12	Axis-5 information	0xaaaa
m+14	Axis-6 information	0xaaaa
m+16	Axis-7 information	0xaaaa
m+18	Axis-8 information	0xaaaa
m+20	Not used	
to		
m+30		

aaaa :Shows the servo status of each axis.

Value	Contents	
0	Servo OFF + mechanical brake ON	(Brake)
1	Servo ON	(Servo)
2	Servo OFF + mechanical brake OFF	(Free)
9	No axis	

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows te additional informaion section in upper 8 bits, and the detail value i lower 8 bits.

Example:

Use this command to acquire a servo status as shown at right.

Channel	Value
n	0x0503
n+2	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Axis 1 : 1 (Servo ON)

Axis 2 : 1 (Servo ON)

Axis 3 : 2 (Servo Free)

Axis 4 : 1 (Servo ON)

Axis 5 : 9 (no axis)

Axis 6 : 9 (no axis)

Axis 7 : 9 no axis)

Axis 8 : 9 (no axis)

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0001
m+6	0x0001
m+8	0x0002
m+10	0x0001
m+12	0x0009
m+14	0x0009
m+16	0x0009
m+18	0x0009
m+20	0x0000
to	
m+30	

4.7.4 Absolute reset status reference

Execute this command to acquire information on absolute reset status.

■ Command

Channel	Contents	Value
n	Command code	0x0504
n+2	Not used	0x0000
to		
n+30		

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
m+4	Axis-1 information	0xaaaa
m+6	Axis-2 information	0xaaaa
m+8	Axis-3 information	0xaaaa
m+10	Axis-4 information	0xaaaa
m+12	Axis-5 information	0xaaaa
m+14	Axis-6 information	0xaaaa
m+16	Axis-7 information	0xaaaa
m+18	Axis-8 information	0xaaaa
m+20	Not used	
to		
m+30		

aaaa :Shows the absolue reset status of each axis.

Value	Contents
0	Return-to-origin incomplete
1	Return-to-origin complete
9	No axis

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shws the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this commnd to acquire an absolute reset status as shown at right.

Channel	Value
n	0x0504
n+2	0x0000
to	
n+30	

Values are exprssed as shown at right when executed correctly.

Axis 1 : 1 (Return-to-origin complete)

Axis 2 : 1 (Return-to-origin complete)

Axis 3 : 0 (Return-to-origin incomplete)

Axis 4 : 1 (Return-to-origin complete)

Axis 5 : 9 (no axis)

Axis 6 : 9 (no axis)

Axis 7 : 9 (no axis)

Axis 8 : 9 (no axis)

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0001
m+6	0x0001
m+8	0x0000
m+10	0x0001
m+12	0x0009
m+14	0x0009
m+16	0x0009
m+18	0x0009
m+20	0x0000
to	
m+30	

4.7.5 Current position reference

Execute this command group to obtain the robot current position data.

4.7.5.1 Pulse designation

Use this command to obtain the robot current position data in pulse units.

■ Command

Channel	Contents		Value
n	Command code	For main robot	0x0505
		For sub robot	0x0585
n+2	Not used		0x0000
to			
n+30			

■ Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	0
		bit 15–bit 1	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

bbbbbbbb :Shows the current position output data in 32 bits. (little endian) Data is shown in integers.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to obtain the main robot current position data in pulse units.

Channel	Value
n	0x0505
n+2	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly to obtain the following positions in pulse units.

Axis 1 = 20001

Axis 3 = -12345

Other axes = 0

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0000
m+8	0x4E21
m+10	0x0000
m+12	0x0000
m+14	0x0000
m+16	0xCFC7
m+18	0xFFFF
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.7.5.2 Millimeter designation

Use this command to obtain the robot current position data in millimeter units.

■ Command

Channel	Contents		Value
n	Command code	For main robot	0x0506
		For sub robot	0x0586
n+2	Not used		0x0000
to			
n+30			

■ Status

Normal end

Channel	Contents			Value
m	Status code			0x0200
m+2	Not used			
m+4				
m+6	Point flag	bit 0	Point unit	1
		bit 2–bit 1	Hand system	tt
		bit 15–bit 3	Not used	0
m+8	Axis-1 data			0xbbbbbbbb
m+10				
m+12	Axis-2 data			0xbbbbbbbb
m+14				
m+16	Axis-3 data			0xbbbbbbbb
m+18				
m+20	Axis-4 data			0xbbbbbbbb
m+22				
m+24	Axis-5 data			0xbbbbbbbb
m+26				
m+28	Axis-6 data			0xbbbbbbbb
m+30				

tt :Shows in 2 bits the current hand system.
Valid only for a SCARA robot is specified.

Value	Meaning
01	Right-handed is specified.
10	Left-handed is specified.

bbbbbbbb :Shows the current position output data in 32 bits. (little endian)
Data is shown in integers (x100).

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to obtain the main robot current position data in millimeter units.

Channel	Value
n	0x0506
n+2	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly to obtain the following positions in millimeter units.

Axis 1 = 200.01

Axis 3 = -123.45

Other axes = 0.00

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0000
m+6	0x0001
m+8	0x4E21
m+10	0x0000
m+12	0x0000
m+14	0x0000
m+16	0xCFC7
m+18	0xFFFF
m+20	0x0000
m+22	0x0000
m+24	0x0000
m+26	0x0000
m+28	0x0000
m+30	0x0000

4.7.6 Task status reference

Execute this command to acquire task execution status.

Command

Channel	Contents	Value
n	Command code	0x0507
n+2	Not used	0x0000
to		
n+30		

Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
m+4	Execution status of task 1	0xaaaa
m+6	Execution status of task 2	0xaaaa
m+8	Execution status of task 3	0xaaaa
m+10	Execution status of task 4	0xaaaa
m+12	Execution status of task 5	0xaaaa
m+14	Execution status of task 6	0xaaaa
m+16	Execution status of task 7	0xaaaa
m+18	Execution status of task 8	0xaaaa
m+20	Not used	
to		
m+30		

aaaa :Shows the execution status of each task.

Value	Contents
0	Stop status
1	Run status (Ready status / Wait status)
2	Suspend status
9	No task

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to acquire the execution status of a task.

Channel	Value
n	0x0507
n+2	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Task 1 : 1 (Run status)

Task 2 : 1 (Run status)

Task 3 : 9 (no task)

Task 4 : 9 (no task)

Task 5 : 2 (Suspend status)

Task 6 : 9 (no task)

Task 7 : 9 (no task)

Task 8 : 9 (no task)

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0001
m+6	0x0001
m+8	0x0009
m+10	0x0009
m+12	0x0002
m+14	0x0009
m+16	0x0009
m+18	0x0009
m+20	0x0000
to	
m+30	

4.7.7 Task execution line reference

Execute this command to acquire information on task execution line.

■ Command

Channel	Contents	Value
n	Command code	0x0508
n+2	Not used	0x0000
to		
n+30		

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
m+4	Execution line of task 1	0xaaaa
m+6	Execution line of task 2	0xaaaa
m+8	Execution line of task 3	0xaaaa
m+10	Execution line of task 4	0xaaaa
m+12	Execution line of task 5	0xaaaa
m+14	Execution line of task 6	0xaaaa
m+16	Execution line of task 7	0xaaaa
m+18	Execution line of task 8	0xaaaa
m+20	Not used	
to		
m+30		

aaaa :Shows the execution line of each task.
When no task exists, the value is 0.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to acquire the execution line of a task.

Channel	Value
n	0x0508
n+2	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Task 1 : Execution on first line

Task 2 : Execution on 19th line

Task 3 : no task

Task 4 : no task

Task 5 : Execution on 99th line

Task 6 : no task

Task 7 : no task

Task 8 : no task

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0001
m+6	0x0013
m+8	0x0000
m+10	0x0000
m+12	0x0063
m+14	0x0000
m+16	0x0000
m+18	0x0000
m+20	0x0000
to	
m+30	

4.7.8 Message reference

Execute this command to acquire information on error message status.

■ Command

Channel	Contents	Value
n	Command code	0x0509
n+2	Not used	0x0000
n+4	Error acquisition number	0xaaaa
n+6	Not used	0x0000
to		
n+30		

aaaa :Specify the error acquisition number.

Value	Contents
0	Message currently displayed on programming box.
1 to 500	Message number stored in error history.

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

NOTE

When "0" is specified for Error acquisition number, the message number and additional information which were last displayed on the programming box will be set to m+2 and m+4 . When error is cleared, 0 will be set to m+2 and m+4.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to acquire the status of an error message.

Channel	Value
n	0x0509
n+2	0x0000
n+4	0x000A
n+6	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0C02
m+4	0x0000
m+6	0x0000
to	
m+30	

4.7.9 Speed status reference

Execute this command to acquire information on current speed status.

■ Command

Channel	Contents	Value
n	Contents	0x050A
n+2	Not used	0x0000
to		
n+30		

■ Status

Normal end

Channel	Contents		Value
m	Status code		0x0200
m+2	Not used		
m+4	Main robot	AUTO mode speed	0xaaaa
m+6		MANUAL mode speed	0xaaaa
m+8	Sub robot	AUTO mode speed	0xaaaa
m+10		MANUAL mode speed	0xaaaa
m+12	Not used		
to			
m+30			

aaaa :Shows the speed setting (1 to 100).
Shows "0" when no robot axis is specified.

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use the speed status reference command as shown at right, to acquire the status of current speed.

Channel	Value
n	0x050A
n+2	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Main robot speed in AUTO mode : 50%

Main robot speed in MANUAL mode : 50%

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0032
m+6	0x0032
m+8	0x0000
m+10	0x0000
m+12	0x0000
to	
m+30	

4.7.10 Arm designation status reference

Execute this command to acquire information on currently designated arm.

■ Command

Channel	Contents	Value
n	Command code	0x050B
n+2	Not used	0x0000
to		
n+30		

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
m+4	Main robot status	0xaaaa
m+6	Sub robot status	0xaaaa
m+8	Not used	
to		
m+30		

aaaa :Shows the arm designation status.

Value	Contents
0	Right-handed system status
1	Left-handed system status
9	Robots other than SCARA robot

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to acquire the status of currently specified arm.

Channel	Value
n	0x050B
n+2	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Main robot : 1 (Left-handed system status)

Sub robot : 9 (Robots other than SCARA robot)

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0001
m+6	0x0009
m+8	0x0000
to	
m+30	

4.7.11 Arm status reference

Execute this command to acquire information on arm.

■ Command

Channel	Contents	Value
n	Command code	0x050C
n+2	Not used	0x0000
to		
n+30		

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
m+4	Main robot status	0xaaaa
m+6	Sub robot status	0xaaaa
m+8	Not used	
to		
m+30		

aaaa :Shows the arm status.

Value	Contents
0	Right-handed system status
1	Left-handed system status
9	Robots other than SCARA robot

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to acquire the status of arm.

Channel	Value
n	0x050C
n+2	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Main robot : 1 (Left-handed system status)

Sub robot : 9 (Robots other than SCARA robot)

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0001
m+6	0x0009
m+8	0x0000
to	
m+30	

4.7.12 Service mode status reference

Execute this command to acquire current information on service mode.
To use this command, DI dedicated input must be enabled by input device exclusive control.

■ Command

Channel	Contents	Value
n	Command code	0x050D
n+2	Not used	0x0000
to		
n+30		

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
m+4	Setting status	0xaaaa
m+6	Not used	
to		
m+30		

aaaa :Shows the service mode setting status.

Value	Contents
0	Normal mode
1	Service mode

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to acquire the status of service mode.

Channel	Value
n	0x050D
n+2	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0001
m+6	0x0000
to	
m+30	

4.7.13 Point unit status reference

Execute this command to acquire current information on point units.

■ Command

Channel	Contents	Value
n	Command code	0x050E
n+2	Not used	0x0000
to		
n+30		

■ Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
m+4	Setting status	0xaaaa
m+6	Not used	
to		
m+30		

aaaa :Shows the point setting status.

Value	Contents
0	Pulse units
1	Millimeter units
2	Tool coordinates

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command as shown at right, to acquire the status of point units.

Channel	Value
n	0x050E
n+2	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0001
m+6	0x0000
to	
m+30	

4.7.14 Return-to-origin status reference

Execute this command to acquire information on the return-to-origin status.

■ Command

Channel	Contents	Value
n	Command code	0x050F
n+2	Not used	0x0000
to		
n+30		

Status

Normal end

Channel	Contents	Value
m	Status code	0x0200
m+2	Not used	
m+4	Axis-1 information	0xaaaa
m+6	Axis-2 information	0xaaaa
m+8	Axis-3 information	0xaaaa
m+10	Axis-4 information	0xaaaa
m+12	Axis-5 information	0xaaaa
m+14	Axis-6 information	0xaaaa
m+16	Axis-7 information	0xaaaa
m+18	Axis-8 information	0xaaaa
m+20	Not used	
to		
m+30		

aaaa :Shows the return-to-origin status of each axis.

Value	Contents
0	Return-to-origin incomplete
1	Return-to-origin complete
9	No axis

Abnormal end

Channel	Contents	Value
m	Status code	0x4000
m+2	Error code	0xaabb
m+4	Additional information	0xccdd
m+6	Not used	
to		
m+30		

aabb :Shows the group number in upper 8 bits, and the category number in lower 8 bits.

ccdd :Shows the additional information section in upper 8 bits, and the detail value in lower 8 bits.

Example:

Use this command to obtain a return-to-origin status as shown at right.

Channel	Value
n	0x050F
n+2	0x0000
to	
n+30	

Values are expressed as shown at right when executed correctly.

Axis 1 : 1 (Return-to-origin complete)

Axis 2 : 1 (Return-to-origin complete)

Axis 3 : 0 (Return-to-origin incomplete)

Axis 4 : 1 (Return-to-origin complete)

Axis 5 : 9 (no axis)

Axis 6 : 9 (no axis)

Axis 7 : 9 (no axis)

Axis 8 : 9 (no axis)

Channel	Value
m	0x0200
m+2	0x0000
m+4	0x0001
m+6	0x0001
m+8	0x0000
m+10	0x0001
m+12	0x0009
m+14	0x0009
m+16	0x0009
m+18	0x0009
m+20	0x0000
to	
m+30	

I/O command guide

Contents

1. I/O command format B-1
2. Sending and receiving I/O commands B-2
3. I/O command list B-3
4. I/O command description B-4

4.1 MOVE command B-4
4.1.1 PTP designation B-4
4.1.2 Linear interpolation B-4
4.2 MOVEI command B-5
4.2.1 PTP designation B-5
4.3 Pallet movement command B-5
4.3.1 PTP designation B-5
4.4 Jog movement command B-6
4.5 Inching movement command B-7
4.6 Point teaching command B-7
4.7 Absolute reset movement command B-8
4.8 Absolute reset command B-8
4.9 Return-to-origin command B-9
4.10 Servo command B-9
4.11 Manual speed change command B-10
4.12 Auto speed change command B-10
4.13 Program speed change command B-11
4.14 Shift designation change command B-11
4.15 Hand designation change command B-11
4.16 Arm designation change command B-12
4.17 Point display unit designation command B-12

1. I/O command format

Using bit information from the SI/SO port on the EtherNet/IP compatible module allows issuing commands directly from the PLC. It is now possible to execute commands such as the MOVE command that were impossible to execute up until now without using the robot program or RS-232C port.

CAUTION

I/O commands are compatible with the RCX240 software version 10.52 or higher.
To use remote commands, the "Remote_cmd SI05" parameter in SYSTEM > PARAM > OP.BRD mode must be set to "VALID" in advance. For details, refer to "3.2. Setting the "Remote_cmd SI05" function" in Chapter 2 of the EtherNet/IP compatible module guide and the user's manual for controller.

The following features are assigned to each I/O.

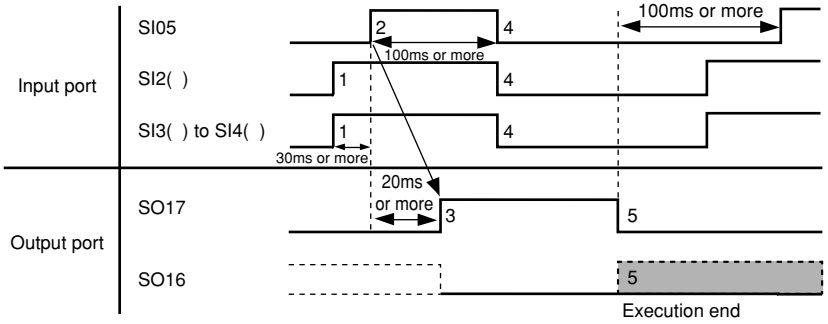
Output (Controller → PLC)		Input (Controller ← PLC)	
Output port	**Contents**	**Input port**	**Contents**
SO16	Execution check output	SI05	I/O command execution trigger input
SO17	Execution in-progress output	SI2()	Command code
		SI3()	Command data
		SI4()	

- I/O commands cannot be executed simultaneously with remote commands.
- Commands cannot be executed unless the status is ready to accept commands (0x0000).
- I/O commands cannot be executed while program execution is in progress (SO13 is ON).
- I/O commands cannot be executed simultaneously with on-line commands.
- I/O commands assign command codes to be executed to SI2(), and command data to SI3() and SI4(). These are executed when the SI05 is changed from OFF to ON. The controller processes the I/O commands when they are received and sends execution check results and execution in-progress information to the PLC via SO16 and SO17.
- Command data added to the I/O commands will differ according to the I/O command. For details, Refer to "4. I/O command description" in this guide. Command data settings must always be made before attempting to set the I/O commands.
- Data is set in binary code. If the data size is greater than 8 bits, set the upper bit data into the higher address. (little endian)
  For example, to set 0x0F9F [hexadecimal] (=3999) in the SI13 () and SI14 () ports, set 0x0F [hexadecimal] in SI4 () and set 0x9F [hexadecimal] in SI13 ().
- The I/O command execution trigger is disabled when the execution in-progress output SO17 is ON.
- The execution in-progress output SO17 is ON in the following cases.
  - When an I/O command is running after receiving I/O command execution trigger input.
  - When an I/O command is terminated after receiving I/O command execution trigger input yet a maximum of 100ms state is maintained when I/O command trigger input is ON.
- The I/O command trigger input pulse must always be maintained for 100ms or more during input. Commands cannot be accepted if this state is not maintained.
- Sometimes 20ms or more is needed for the execution in-progress output SO17 to turn ON after startup (rising edge) of the I/O command trigger input pulse. The I/O command trigger input might not be accepted during this period.
- After inputting the I/O command trigger input pulse and the in-progress output turns OFF, at least a 100ms time period must always elapse before executing the next command. If this elapsed time period is too small, the I/O command execution trigger input might not be accepted.
- The execution check output SO16 turns OFF when an I/O command is received.
- The execution check output SO16 turns ON when an I/O command ended correctly, but stays OFF if an I/O command ended abnormally.

2. Sending and receiving I/O commands

Sending and receiving is performed in the I/O register as shown below.

Sending and receiving I/O commands



93301-S6-00

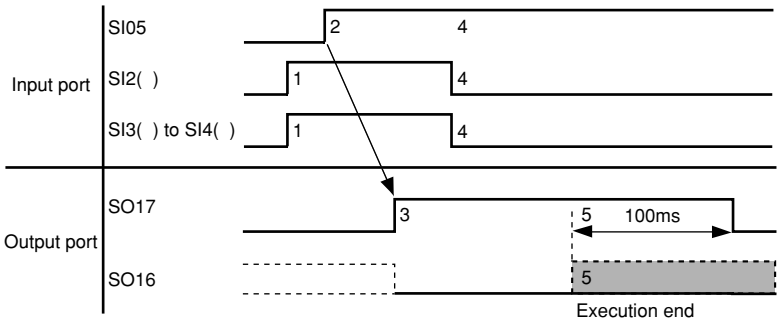
1. Set command code and command data (Time interval between 1 and 2: 30ms or more)
2. Set I/O command execution trigger input (Pulse width: 100ms or more)
3. Transition to execute
4. Clear the I/O command trigger input and command code and command data
5. Set termination of I/O command and execution check output

Example :Follow these steps when sending and receiving I/O commands to execute the PTP movement command to point 19.

1. Set the following values in the register to execute the PTP movement command by designating a point.
   SI2 ( ) : Command code (0x01)
   SI3 ( ) : Lower point setting (0x13= point 19)
   SI4 ( ) : Upper point setting (0x00=point 19)
2. Set to ON after SI05 turns OFF.
3. The controller receives the I/O command and executes it if the command and command data are acceptable. The in-progress output (SO17) turns ON and the execution check output (SO16) is turned OFF at this time. The robot moves to the position specified by point 19.
4. Clear SI2 ( ) through SI4 ( ) after checking that in-progress output (SO17) is ON.
5. The command has ended so in-progress output (SO17) turns OFF and execution check output (SO16) turns ON if command ended correctly, and turns OFF if the command failed.

* If SI05 was not set to OFF in 4, the in-progress output (SO17) remains ON for a maximum of 100ms from the timing in 5.

Example of sending and receiving



93302-S6-00

3. I/O command list

I/O commands are expressed with hexadecimal codes.

No.	Command contents		Command code (SI2( ))	
			Main robot	Sub robot
1	MOVE command	PTP point designation	0x01	0x81
		Linear interpolation	0x03	0x83
2	MOVEI command	PTP designation	0x09	0x89
3	Pallet movement command	PTP designation at pallet 0	0x18	0x98
4	Jog movement command		0x20	0xA0
5	Inching movement command		0x24	0xA4
6	Point teaching command		0x28	0xA8
7	Absolute reset movement command		0x30	0xB0
8	Absolute reset command		0x31	0xB1
9	Return-to-origin command		0x32	0xB2
10	Servo command	On designation	0x34	0xB4
		Off designation	0x35	0xB5
		Free designation	0x36	0xB6
		Power-on designation	0x37	
11	Manual movement speed command		0x38	0xB8
12	Auto movement speed command		0x39	0xB9
13	Program speed change command		0x3A	0xBA
14	Shift designation change command		0x3B	0xBB
15	Hand designation change command		0x3C	0xBC
16	Arm designation change command		0x3D	0xBD
17	Point display unit designation command		0x3E	

* The pallet movement command (3) is only valid for pallet 0.
* The movement methods on the jog movement command (4) and inching movement command (5) will differ according to the point units that were specified.
* The point teaching command (6) uses different point units according to the point units that were specified.
* If no axis is specified, the absolute reset command (8) is executed on all axes (main robot + sub robot) in either case of command code 0x31 or 0xB1.
* If no axis is specified, the return-to-origin command (9) is executed on all axes (main robot + sub robot) in either case of command code 0x32 or 0xB2.
* The point display unit designation command (17) is for use on the controller.

4. I/O command description

4.1 MOVE command

Execute this command group to move the robot to an absolute position.

4.1.1 PTP designation

This command moves the robot to a target position in PTP motion by specifying the point number.

■ **Command**

SI port	Contents		Value
SI2()	Command code	For main robot	0x01
		For sub robot	0x81
SI3()	Point number		0xpppp
SI4()			

pppp :Specify the point number in 16 bits.
Specified range: 0 (=0x0000) to 9999 (=0x270F)

NOTE

When a SCARA robot is used and a hand system flag is set for the point data you specify, that hand system has priority over the current arm type.

4.1.2 Linear interpolation

This command moves the robot to a target position by linear interpolation by specifying the point number.

■ **Command**

SI port	Contents		Value
SI2()	Command code	For main robot	0x03
		For sub robot	0x83
SI3()	Point number		0xpppp
SI4()			

pppp :Specify the point number in 16 bits.
Specified range: 0 (=0x0000) to 9999 (=0x270F)

CAUTION

- **The point number setting range is 0 to 255 when there is no SI4().**
- **When moving the robot by linear interpolation to a point where a hand system flag is specified, make sure that the same hand system is used at the current position and target position. If the same hand system is not used, then an error will occur and robot movement will not be allowed.**

NOTE

When a SCARA robot is used and a hand system flag is set for the point data you specify, that hand system has priority over the current arm type.

4.2 MOVEI command

Execute this command group to move the robot to a relative position.

4.2.1 PTP designation

This command moves the robot a specified distance in PTP motion by specifying the point number.

NOTE

- If the MOVEI command is interrupted and then re-executed, the resumed motion that occurs either to the original target position or to a new target position referenced to the current position can be selected by the "MOVEI/DRIVEI start position" setting of other parameters. For details, refer to the controller user's manual.
- The other parameters default "MOVEI/DRIVEI start position" setting is Keep (motion to the original target position when MOVEI is interrupted and then re-executed).

■ Command

SI port	Contents		Value
SI2()	Command code	For main robot	0x09
		For sub robot	0x89
SI3()	Point number		0xpppp
SI4()			

pppp :Specify the point number in 16 bits.
Specified range: 0 (=0x0000) to 9999 (=0x270F)

CAUTION

The point number setting range is 0 to 255 when there is no SI4().

NOTE

When a SCARA robot is used and a hand system flag is set for the point data you specify, that hand system has priority over the current arm type.

4.3 Pallet movement command

Execute this command group to move the robot to a position with respect to pallet 0.

4.3.1 PTP designation

This command moves the robot to a target position in PTP motion by specifying the work position number.

■ Command

SI port	Contents		Value
SI2()	Command code	For main robot	0x18
		For sub robot	0x98
SI3()	Work position number		0xwwww
SI4()			

wwww :Specify the work position number in 16 bits.
Specified range: 1 (=0x0001) to 32767 (=0x7FFF)

CAUTION

If SI4() is not present, then the specified range of the work position number will be 0 to 255.

4.4 Jog movement command

This command moves the robot in jog mode while in MANUAL mode. This command is only valid in MANUAL mode. This command is linked with the controller point display units. The robot axis moves in PTP motion when display units are in pulses, and moves by linear interpolation on Cartesian coordinates when units are in millimeters. Jog speed is determined by the MANUAL speed.

To stop the jog movement command, set the dedicated input interlock signal (SI11) to OFF. After checking that jog movement has stopped, set the interlock signal back to ON.

■ Command

SI port	Contents			Value
SI2()	Command code		For main robot	0x20
			For sub robot	0xA0
SI3()	Axis to move and direction	bit 0	Axis 1	tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 6	(0:Fixed)	0
		bit 7	Direction	d
SI4()	Not used			0x00

tt :Specify the axis to move in 0 to 3 bits.
Only one axis can be specified.

d :Specify the movement direction in 1 bit.

Value	Meaning
0	+ direction
1	- direction

4.5 Inching movement command

Execute this command to move the robot by inching in MANUAL mode. Inching movement distance is linked to the manual movement speed. The inching command can only be executed in MANUAL mode.
This command is linked with the controller's point display unit system. So when display units are in pulses, the axis moves a certain number of pulses at the manual speed setting. When display units are in millimeters, the axis moves on Cartesian coordinates by linear interpolation at the manual speed setting divided by 100.

■ Command

SI port	Contents			Value
SI2()	Command code		For main robot	0x24
			For sub robot	0xA4
SI3()	Axis to move and direction	bit 0	Axis 1	tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 6	(0:Fixed)	0
		bit 7	Direction	d
SI4()	Not used			0x00

tt :Specify the axis to move in 0 to 3 bits.
Only one axis can be specified.

d :Specify the movement direction in 1 bit.

Value	Meaning
0	+ direction
1	- direction

4.6 Point teaching command

Execute this command to teach the current robot position to the specified point number.
Point data units of this command are linked to the controller's point display unit system.

■ Command

SI port	Contents		Value
SI2()	Command code	For main robot	0x28
		For sub robot	0xA8
SI3()	Point number		0xpppp
SI4()			

pppp :Specify the point number in 16 bits.
Specified range: 0 (=0x0000) to 9999 (=0x270F)

4.7 Absolute reset movement command

When absolute reset of the specified axis uses the mark method, this command moves the axis to the nearest position where absolute reset can be executed. Positions capable of absolute reset are located at every 1/4 rotation of the motor.

■ Command

SI port	Contents			Value
SI2()	Command code		For main robot	0x30
			For sub robot	0xB0
SI3()	Axis to move and direction	bit 0	Axis 1	tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 6	(0:Fixed)	0
		bit 7	Direction	d
SI4()	Not used			0x00

tt :Specify the axis to move in 0 to 3 bits.
Only one axis can be specified.

d :Specify the movement direction in 1 bit.

Value	Meaning
0	+ direction
1	- direction

4.8 Absolute reset command

This command executes absolute reset of the specified axis. When absolute reset of the specified axis uses the mark method, the axis must be at a position where absolute reset can be executed. If no axis is specified (SI3() is 0), then absolute reset is performed on all axes (main robot + sub robot) in either case of command code 0x31 or 0xB1. However, this command cannot be executed if return-to-origin is not yet complete on the axis using the mark method. In this case, perform return-to-origin individually on each axis.

■ Command

SI port	Contents			Value
SI2()	Command code		For main robot	0x31
			For sub robot	0xB1
SI3()	Specified axis	bit 0	Axis 1	tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 7–bit 6	(0:Fixed)	0
SI4()	Not used			0x00

tt :Specify the axis to perform absolute reset in 0 to 3 bits.
Only one axis can be specified.
If no particular axis is specified then absolute reset is performed on all axes (main robot + sub robot).

4.9 Return-to-origin command

This command executes return-to-origin on the specified axis.
When this command is executed, return-to-origin is performed on an incremental mode axis and absolute search is performed on a semi-absolute mode axis. If no axis is specified (SI3() is 0), this command is performed on all axes (main robot + sub robot) in either case of command code 0x32 or 0xB2.

■ Command

SI port	Contents			Value
SI2()	Command code		For main robot	0x32
			For sub robot	0xB2
SI3()	Specified axis	bit 0	Axis 1	tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 7–bit 6	(0:Fixed)	0
SI4()	Not used			0x00

tt :Specify the axis to perform return-to-origin in 0 to 3 bits.
Only one axis can be specified.
If no particular axis is specified then return-to-origin is performed on all axes (main robot + sub robot).

4.10 Servo command

Execute this command group to operate the robot servos.

Servo ON :
Execute this command to turn on the servo of a specified axis. The motor power must be turned on when specifying the axis. All controller servos are turned on if no axis is specified.

Servo OFF :
Execute this command to turn off the servo of a specified axis. All controller servos are turned off if no axis is specified.

Servo Free :
Execute this command to turn off the mechanical brake and dynamic brake after turning off the servo of a specified axis. Servo OFF and Free are repeated when this command is consecutively executed.

Power ON:
Execute this command to turn on the motor power. No axis can be specified.

Command

SI port	Contents			Value
SI2()	Command code	Servo ON	For main robot	0x34
			For sub robot	0xB4
		Servo OFF	For main robot	0x35
			For sub robot	0xB5
		Servo Free	For main robot	0x36
			For sub robot	0xB6
		Power ON	All controller servos	0x37
SI3()	Specified axis	bit 0	Axis 1	tt
		bit 1	Axis 2	
		bit 2	Axis 3	
		bit 3	Axis 4	
		bit 4	Axis 5	
		bit 5	Axis 6	
		bit 7–bit 6	(0:Fixed)	0
SI4()	Not used			0x00

tt :Specify the axis to move in 0 to 3 bits. All controller servos are processed if no axis is specified. No axis can be specified when executing Power ON.

4.11 Manual speed change command

Execute this command to change the manual movement speed in MANUAL mode. This command can only be executed in MANUAL mode.

Command

SI port	Contents		Value
SI2()	Command code	For main robot	0x38
		For sub robot	0xB8
SI3()	Specified speed		0xss
SI4()	Not used		0x00

ss :Specify the manual movement speed in 8 bits.
Specified range: 1 (=0x01) to 100 (=0x64)

4.12 Auto speed change command

Execute this command to change the auto movement speed in AUTO mode.

Command

SI port	Contents		Value
SI2()	Command code	For main robot	0x39
		For sub robot	0xB9
SI3()	Specified speed		0xss
SI4()	Not used		0x00

ss :Specify the auto movement speed in 8 bits.
Specified range: 1 (=0x01) to 100 (=0x64)

4.13 Program speed change command

Execute this command to change the program speed in AUTO mode. The program speed changed with this command is reset to 100% when the program is reset or changed.

■ Command

SI port	Contents		Value
SI2()	Command code	For main robot	0x3A
		For sub robot	0xBA
SI3()	Specified speed		0xss
SI4()	Not used		0x00

ss :Specify the program speed in 8 bits.
Specified range: 1 (=0x01) to 100 (=0x64)

4.14 Shift designation change command

Execute this command to change the selected shift to a specified shift number.

■ Command

SI port	Contents		Value
SI2()	Command code	For main robot	0x3B
		For sub robot	0xBB
SI3()	Specified shift number		0xss
SI4()	Not used		0x00

ss :Specify the shift number in 8 bits.
Specified range: 0 (=0x00) to 9 (0x09)

4.15 Hand designation change command

Execute this command to change the selected hand to a specified hand number.

■ Command

SI port	Contents		Value
SI2()	Command code	For main robot	0x3C
		For sub robot	0xBC
SI3()	Specified hand number		0xss
SI4()	Not used		0x00

ss :Specify the hand number in 8 bits.
Specified range for main robot : 0 (=0x00) to 3 (0x03)
Specified range for sub robot : 4 (=0x04) to 7 (0x07)

4.16 Arm designation change command

Execute this command to change the arm designation status.

■ Command

SI port	Contents		Value
SI2()	Command code	For main robot	0x3D
		For sub robot	0xBD
SI3()	Status of specified arm		0xss
SI4()	Not used		0x00

ss :Specify the arm designation status in 8 bits.

Value	Meaning
0x00	Right-handed system
0x01	Left-handed system

4.17 Point display unit designation command

Execute this command to change the point display unit.

■ Command

SI port	Contents		Value
SI2()	Command code	For main robot	0x3E
		For sub robot	
SI3()	Display units for specified point		0xss
SI4()	Not used		0x00

ss :Specify the point display unit in 8 bits.

Value	Meaning
0x00	Pulse units
0x01	Millimeter units
0x02	Millimeter units (Tool coordinates)

Revision record

Manual version	Issue date	Description
Ver. 1.00	Feb. 2012	First edition

User's Manual

YAMAHA
Robot Controller

RCX240
EtherNet/IP
network board

Feb. 2012
Ver. 1.00
This manual is based on Ver. 1.00 of Japanese manual.

© YAMAHA MOTOR CO., LTD.
IM Operations

All rights reserved. No part of this publication may be reproduced in any form without the permission of YAMAHA MOTOR CO., LTD.
Information furnished by YAMAHA in this manual is believed to be reliable. However, no responsibility is assumed for possible inaccuracies or omissions. If you find any part unclear in this manual, please contact YAMAHA or YAMAHA sales representatives.

IM Operations

882 Soude, Nakaku, Hamamatsu, Shizuoka, 435-0054, Japan
Tel. 81-53-460-6103 Fax. 81-53-460-6811

Instruction manuals can be downloaded from our company website.
Please use the following for more detailed information.
http://www.yamaha-motor.co.jp/global/industrial/robot/

YAMAHA MOTOR CO., LTD.